UNIVERSIDAD CARLOS III DE MADRID

TESIS DOCTORAL

METODOLOGÍA PARA LA VALIDACIÓN Y EVALUACIÓN REMOTA DE IMPLEMENTACIONES DE PROTOCOLOS DE SEGURIDAD. APLICACIÓN A LA ARQUITECTURA IPSEC

Autor:
Antonio Izquierdo Manzanares

Director:
Jose María Sierra Cámara

DEPARTAMENTO DE INFORMÁTICA

Leganés, Octubre de 2.006

Resumen

Las redes de comunicaciones han pasado a ser parte fundamental de las tecnologías de la información, siendo el medio a través del cual los sistemas informáticos de todo tipo (desde ordenadores personales hasta cajeros automáticos y ordenadores de a bordo en los coches) intercambian la información que necesitan para llevar a cabo las tareas para las que han sido diseñados. Estos intercambios de información están regidos por protocolos de comunicaciones que gobiernan la forma en la que diferentes entidades proceden a enviarse la información de la forma más eficiente y conveniente posible. En muchas ocasiones estos intercambios de información requieren de servicios de seguridad (como son la confidencialidad, la autenticación o el no repudio) de los que carecen los protocolos diseñados al comienzo de la expansión de las redes de comunicaciones (protocolos que son los más extendidos y utilizados en la actualidad). Para cubrir este vacío de seguridad se han diseñado y estandarizado protocolos y arquitecturas de seguridad que proporcionan los servicios de seguridad requeridos al resto de protocolos de comunicaciones. La arquitectura de seguridad IPsec está ampliamente extendida debido a su transparencia de cara a aplicaciones y usuarios, y a que su integración en los protocolos de red de siguiente generación garantiza que la migración se podrá llevar a cabo de la forma menos traumática posible. Sin embargo, el uso de estos protocolos y arquitecturas de seguridad ha ocasionado la aparición de nuevos problemas, entre los que se pueden destacar los problemas de interoperabilidad entre las diferentes implementaciones de los protocolos y arquitecturas, o una mayor predisposición a sufrir ataques de denegación de servicio. Con el objetivo de ofrecer información acerca de la implementación que ayude a evitar este tipo de problemas, en esta tesis se propone una metodología que permita evaluar el nivel de conformidad de una implementación de un protocolo o arquitectura de seguridad con el estándar y el rendimiento que dicha implementación puede ofrecer. La metodología propuesta se desarrolla para aplicarse a la arquitectura de seguridad IPsec con el fin de demostrar la aplicación práctica de la propuesta, y se presentan los resultados obtenidos mediante la evaluación de diferentes implementaciones de IPsec.

Abstract

Communication networks have become a key component of information technologies, being the means through which computer systems all around the world, independently of their nature (from personal computers to ATMs, and on-board computers in high-end cars) share the information they need in order to accomplish the tasks they are charged with. These information exchanges are carried out following the guidelines of communications protocols, which guide and direct the way in which several entities exchange information in the most efficient and convenient way. As it happens more often than not, these information exchanges require some security services to be present (such as confidentiality, authentication or non-repudiation) which are not included in protocols designed in the early years of communication networks (those same protocols that are now widely spread and used nowadays). In order to provide with a solution for this need for security, security protocols and architectures which provide security services to other network protocols have been standardized. The IPsec security architecture is widely spread thanks to its seamless integration with applications and users, and also due to its integration with next-generation communication protocols provides a means for migrating to those new protocols in a safe and more convenient way. However, with the use of these security protocols and architectures new problems have emerged, some of which are the interoperability issues among different implementations of those protocols and architectures, or an increased chance to suffer denial of service attacks. In order to gather information about those implementations that helps to prevent these kind of problems, in this thesis a new methodology that allows the evaluation of the conformance with the standard, as well as the performance of such implementation is proposed. The methodology is applied to the IPsec security architecture in order to demonstrate the practical application of our proposal, and the results obtained when evaluating several IPsec implementations are presented.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

1.1. Origen de la tesis

En los últimos años hemos visto cómo las redes de comunicaciones han pasado a ser parte de nuestra vida cotidiana. A medida que el tiempo ha pasado, las redes de ordenadores han dejado de ser un medio de comunicación para un segmento específico de la población para pasar a ser utilizadas por la mayoría de los ciudadanos y empresas, utilizándose para actividades que varían desde la compra electrónica hasta el control logístico en puertos y aeropuertos internacionales. Este incremento en el uso de las redes de comunicaciones ha tenido múltiples consecuencias, pudiendo destacar entre ellas el aumento de la cantidad y la importancia de la información que fluye por estas redes. En muchos casos nos encontramos con que esta información requiere de algún tipo de protección en su tránsito por las diferentes redes, ya sea mediante servicios de confidencialidad, de autenticación, etc.... Además, el aumento en el uso de las redes de comunicaciones ha llevado aparejado un incremento en el tipo de dispositivos que pueblan dichas redes: desde supercomputadores y ordenadores pertenecientes a múltiples empresas hasta teléfonos móviles, agendas personales e incluso consolas y plataformas de juegos, llegando al extremo en el que el término "*redes de ordenadores*" ha sido desplazado por el de "*redes de comunicaciones*".

Esta popularización de las redes de comunicaciones ha llevado aparejada el incremento de información que se transmite por ellas, y actualmente es habitual utilizarla para operaciones tan comunes como llevar a cabo transacciones con la Administración pública (por ejemplo, la presentación de la declaración de impuestos), adquirir entradas para múltiples espectáculos culturales o deportivos, e incluso descargar películas directamente a la televisión de nuestro hogar. Sin embargo, otra consecuencia de esta popularización ha sido el incremento de actividades delictivas llevadas a cabo utilizando estas redes. Dichas actividades delictivas se presentan de muchas

formas, desde actualizaciones de estafas y engaños tradicionales hasta la completa suplantación de identidad gracias a información personal recabada de la víctima.

Debido a esta proliferación de ataques y al ya mencionado incremento de la información que se transmite a través de las redes de comunicaciones, es necesario proporcionar los medios necesarios para proteger a los usuarios y sus datos de estos ataques. De manera general, las medidas de seguridad pueden clasificarse en medidas que protegen los equipos o dispositivos, y medidas que protegen la comunicación en sí misma. Las primeras están orientadas a proteger de ataques que permitan obtener o destruir la información que en ellos se almacena (mediante, por ejemplo, virus o "caballos de Troya") el sistema, equipo o dispositivo que se utiliza para acceder a la red o que proporciona un servicio determinado en esa red. Tradicionalmente estas medidas de seguridad se llevan a cabo instalando algún tipo de software o hardware en el sistema, que se encargará de proporcionar las herramientas y medidas necesarias.

Por el contrario, las medidas de seguridad que protegen la comunicación tienen por objetivo hacer que la transmisión de información a través de redes de comunicaciones se lleve a cabo de la manera más segura posible, proporcionando herramientas para poder asegurar que el equipo o sistema al que se está enviando (o del que se está recibiendo información) es aquél con el que se desea intercambiar la información, para prevenir el acceso no autorizado por parte de terceras personas a la información en tránsito, para asegurar que la información no es modificada (accidental o intencionadamente) durante su transmisión, etc.... La protección de la comunicación se lleva a cabo mediante el uso de protocolos de seguridad que se integran en la pila de protocolos que utilizan los extremos de la comunicación. Dependiendo del nivel de la pila al que se realice la integración, será necesario hacer que las aplicaciones o el sistema operativo incluyan las operaciones correspondientes al protocolo de seguridad.

Dado que el diseño de estos protocolos de seguridad no es una tarea trivial y su realización incorrecta puede poner en peligro la seguridad de todo un sistema si el protocolo resultante es defectuoso, el diseño y análisis de estos protocolos se ha dejado en manos de los organismos encargados de diseñar y estandarizar el resto de los protocolos de comunicaciones. En los últimos años, múltiples protocolos de seguridad se han estandarizado, de forma que las especificaciones de estos mecanismos para proteger la información estuviesen disponibles para cualquier grupo de desarrollo. Algunos de estos estándares que se encuentran ampliamente integrados con las herramientas de comunicaciones en la actualidad son *Transport Layer Security* (TLS, [49]), *IPsec* [90],[91] y todos sus protocolos internos (*Internet Key Exchange*, IKE [65], [25], *Authentication Header*, AH [88], [86]), *Kerberos* (en su versión 5, [107]), *Secure Shell* (SSH, [154]), *Extensible Authentication*

Protocol (EAP, [1]), etc. . . ., y muchos otros. Esta situación ha llevado aparejada la aparición de múltiples fabricantes con productos que proporcionan mecanismos para asegurar las redes de comunicaciones. El número y tipo de soluciones de seguridad que podemos encontrar hoy en día para proteger nuestra red son muy elevados, disponiendo de soluciones hardware, software, integradas con el sistema operativo o completamente independientes, utilizando criptografía fuerte o ligera, etc. . . .

Sin embargo, la competencia en este mercado ha llevado a los fabricantes a tener que 'mejorar' su solución para disponer de una ventaja competitiva frente al resto, y eso ha hecho que las soluciones de seguridad hayan sido dotadas de características adicionales: mejoras en el rendimiento y mayor variedad de suites criptográficas son algunas de las 'ventajas' más comunes que podemos encontrar hoy en día en las soluciones de seguridad. Por desgracia, estas mejoras y modificaciones a las soluciones de seguridad han conducido en muchos casos a modificaciones en la forma en la que el estándar se implementa en la solución de seguridad, haciendo que los dispositivos o programas que incorporan estas "mejoras" se vean afectados por múltiples problemas de interoperabilidad cuando intentan establecer un canal de comunicaciones seguro con implementaciones de distinto fabricante, las cual no incorpora las modificaciones llevadas a cabo sobre el estándar, como podemos ver en [99], donde podemos encontrar una lista de implementaciones de IPsec que pueden establecer túneles criptográficos con la implementación OpenS/WAN, y en [45], donde se citan las implementaciones que han pasado una prueba de interoperabilidad desarrollada por los autores.

Esta situación se está convirtiendo en algo cada vez más corriente, debido por un lado a la multiplicidad de fabricantes de soluciones de seguridad para las redes de comunicaciones, y por otro a la creciente dificultad que es la programación en dispositivos con un elevado nivel de heterogeneidad (que incluye los lenguajes de programación utilizados, los recursos disponibles, etc. . .).

De manera general, la mayor parte de las desviaciones del estándar que podemos encontrar en las implementaciones de protocolos de seguridad estandarizados tienen su origen en alguna de las siguientes causas:

- No utilización de algoritmos criptográficos estandarizados, o implementación deficiente de los mismos ([31], [32])
- Programación defectuosa al implementar las especificaciones del estándar ([33], [30])
- Modificaciones y optimizaciones que los fabricantes realizan sobre lo establecido en el estándar para aumentar el valor competitivo de sus productos ([34], [29])

Las consecuencias en el ámbito de la seguridad de estos problemas son

más graves de lo que en un inicio se podría pensar, por dos motivos principales:

1. **El problema se produce en el software o hardware cuya función es proteger la información**. Si el funcionamiento de este hardware o software es irregular se está provocando en el usuario del mismo una falsa sensación de seguridad que podría poner en un riesgo innecesario la información hasta que el problema sea detectado y solucionado.

2. **Otras herramientas o mecanismos de seguridad pueden sufrir las consecuencias**. Un funcionamiento deficiente de una herramienta de seguridad al, por ejemplo, negociar los parámetros criptográficos que se utilizarán posteriormente para proteger la confidencialidad de la información transmitida, podría hacer que las políticas de seguridad destinadas a proteger la información se vean vulneradas inconscientemente. Adicionalmente, si nuestro problema es incorrectamente identificado por terceras partes como un intento de utilizar soluciones de seguridad de baja calidad, podría ocurrir que mediante redes de confianza esa "reputación" se transmita, impidiéndonos utilizar otros servicios necesarios para el correcto desempeño de las labores de nuestra empresa o persona.

Una de las arquitecturas de seguridad que mayores problemas de interoperatividad presenta es IPsec. Uno de los motivos que se suelen utilizar para justificar esta situación es la constante evolución y revisión que sufre la arquitectura. Como ejemplo, cabe decir que en la actualidad existen dos versiones estandarizadas de IPsec: la original, de 1.998 y una más reciente de Diciembre de 2.005 que se encuentra en estado de PROPOSED STANDARD en el IETF. Esta nueva versión no modifica las capacidades ni las características de operación de la arquitectura IPsec, sino que pasa a incluir en el estándar algunos aspectos que eran opcionales en aquella primera versión (como la notificación de congestión de la red (ECN, Explicit Congestion Notification) o se modifican los mecanismos para solucionar problemas o situaciones especiales (como el NAT traversal, con el que se ofrece una solución para que la modificación de la dirección de red de los equipos se conectan a la red no afecte a las propiedades de seguridad ni a los dispositivos que utilizan NAT). Otro de los cambios importantes que se han producido en esta versión es el agrupamiento de todos los protocolos y mecanismos de gestión de claves (IKEv1, ISAKMP, etc. . . .) en IKEv2, con lo que la secuencia de mensajes utilizados ha sido revisada y actualizada. Por este motivo las dos versiones que se pueden encontrar hoy en día de IPsec ofrecen la misma funcionalidad aunque no son compatibles entre sí.

Sin embargo, una de las causas de esta deficiente interoperatividad entre las implementaciones de IPsec es la escasez de metodologías que puedan

informar acerca de las capacidades de interoperatividad de una implementación dada. Las metodologías existentes se centran en analizar un sistema de información en un entorno concreto y determinado, pero sin evaluar el nivel de cumplimiento de los requisitos (que en el caso de una implementación de los protocolos de seguridad, vendrían dados por el estándar del protocolo). Esto quiere decir que muchos de los análisis que plantean no tienen sentido o no son aplicables a la implementación de un protocolo de seguridad (por ejemplo, el control de acceso). Esta claro que como desarrollos que son, las implementaciones de los protocolos de seguridad siempre podrán someterse a un análisis por parte de una de estas metodologías, pero el haber obtenido un nivel de seguridad determinado no implica que la implementación sea conforme al estándar, y por lo tanto, **no asegura la interoperatividad con otros dispositivos**. Sin embargo, es posible encontrar proyectos focalizados en la interoperatividad de las implementaciones, como es el caso del proyecto IPsec-WIT, del National Institute of Standards and Technology (NIST) estadounidense ([120]).

1.2. Interoperatividad

En el ámbito de la interoperatividad, las consecuencias de que las implementaciones, ya sean en hardware dedicado o en software que se integre con el sistema operativo, no se ajusten al estándar se pueden resumir en la imposibilidad de conectarnos con redes que utilicen dispositivos diferentes al nuestro. Por un lado esto es un problema económico (ya que una empresa quedaría ligada al proveedor de esa solución a menos que pueda permitirse desechar todos los equipos o licencias adquiridas y comprar otras nuevas de otro fabricante, las cuales pueden llevar asociado el mismo problema), pero por otro lado es un obstáculo importante al desarrollo de las redes de comunicaciones.

El alcance de estos problemas ya ha sido detectado por los responsables de la investigación tanto en Europa como en Estados Unidos, y así, en el VII Programa Marco de la Unión Europea, se proponen varias áreas de interés con un ámbito de trabajo importante para el desarrollo de la seguridad, una de las cuales se centra en la "*Integración e Interoperatividad de los Sistemas de Seguridad*" ([126]). Las acciones que se lleven a cabo en este área deben estar orientadas a contribuir al desarrollo de otras propuestas e intereses al proporcionar los medios para asegurar y validar la interoperatividad de esos sistemas. De esta manera, las actividades fomentadas desde este área se centrarán en estudiar, evaluar y mejorar la interoperatividad e intercomunicación de sistemas, equipos, servicios y procesos de seguridad.

A través de esta área de interés, la Unión Europea ha reconocido la existencia del problema de interoperatividad en los mecanismos para prote-

ger las redes de comunicaciones, y pretende proporcionar las herramientas, mecanismos y diseños necesarios para evitarlos. Además, en este VII Programa Marco también se desea afrontar la definición de metodologías que se lleven a cabo para evitar la falta de continuidad de los resultados que se alcancen.

Adicionalmente, el desarrollo de los *ecosistemas digitales* (cuyo inicio se sitúa en el VI Programa Marco, siendo continuado en el VII) tiene también importantes implicaciones en lo que respecta a la necesidad de asegurar que diferentes sistemas, mecanismos y soluciones sean capaces de operar entre ellos sin especiales dificultades. Dado que los ecosistemas digitales abarcan desde pequeñas y medianas empresas hasta grandes agrupaciones internacionales de compañías, para la consecución de los objetivos propuestos hay que proporcionar una plataforma universal y compuesta de estándares, de manera que cualquier componente del ecosistema digital pueda hacer uso de ella, independientemente de los dispositivos o tecnologías propias que utilice para acceder a la plataforma.

Por todos estos motivos podemos decir que el VII Programa Marco es un respaldo importante a la investigación recogida en esta tesis doctoral, ya que demuestra que la necesidad de evaluar y asegurar la interoperatividad de los sistemas, especialmente en el área de seguridad, ha sido identificada por otros organismos y centros de investigación internacionales, al tiempo que deja entrever que las metodologías de evaluación y validación existentes no ofrecen solución alguna para estos problemas.

Como hemos podido observar, los protocolos y arquitecturas de seguridad se encuentran en la actualidad en una situación delicada, ya que, de proseguir la tendencia actual, la futura interoperatividad entre los diferentes sistemas que utilicen las redes de comunicaciones tendrá que llevarse a cabo sin mecanismos que garanticen los requisitos de seguridad necesarios.

1.3. Rendimiento de los protocolos de seguridad

Además de la interoperatividad y la conformidad con los estándares, también existen otras características de las implementaciones que es necesario conocer de cara a diseñar otras soluciones de seguridad globales o con un ámbito de operación mayor que el de la propia implementación. Por ejemplo, lad herramientas destinadas a la detección de ataques de denegación de servicio requieren del conocimiento previo acerca de las capacidades que puede ofrecer un dispositivo de red (por ejemplo, ancho de banda), con el fin de detectar cuándo los recursos disponibles están siendo sobreutilizados sistemáticamente. Ejemplos de herramientas similares propuestas por la comunidad científica las podemos encontrar en [135], donde la aplicación de técnicas y mecanismos relacionados con la *calidad de servicio* (QoS) per-

mite gestionar de forma eficiente los recursos, o en [153], donde se utilizan técnicas similares para prevenir denegaciones de servicio (DoS). Este tipo de información no sólo ayuda a implementar de forma más eficiente los mecanismos de seguridad que protegerán nuestra red, sino que, desde el punto de vista de la investigación, un mejor conocimiento del impacto de los protocolos de seguridad en las capacidades de computación y comunicación de un dispositivo o equipo ayudará a introducir mecanismos de seguridad en otro tipo de comunicaciones que actualmente se encuentran desprotegidas, como las comunicaciones de voz sobre IP (VoIP, [95]), o los terminales móviles ([13]).

Desde un punto de visto práctico, el conocimiento del rendimiento que puede ofrecer una solución de seguridad es importante para, principalmente, evitar ocasionar una denegación de servicio a nuestra propia red o sistema (como se describe en [5]) al exigir de la solución de seguridad un rendimiento mayor del que puede ofrecer, al tiempo que sirve de baremo comparativo entre las diferentes soluciones que los proveedores de seguridad pueden ofrecer. Un ejemplo de este tipo de situaciones lo podemos ver en los teléfonos móviles y agendas electrónicas disponibles hoy en día: En estos dispositivos se nos suele ofrecer la posibilidad de cifrar los datos que almacenamos en ellos (por ejemplo, contactos, calendario, citas, etc. . . .), pero si tenemos almacenada una gran cantidad de información (entendiendo por "gran cantidad" un valor relativo a las capacidades de este tipo de dispositivos, como puede ser más de 5 MBytes), el rendimiento de dichos dispositivos al realizar las operaciones de cifrado y descifrado se reduce drásticamente.

En general, la vulnerabilidad de los sistemas a ataques de este tipo es muy común, especialmente en los dispositivos móviles. Por lo tanto, es extremadamente factible el llevar a cabo una denegación de servicio contra un determinado dispositivo o sistema que utilice de forma poco apropiada los mecanismos y herramientas de seguridad (como se puede ver en [79], donde los recursos que requiere la técnica de control de acceso *Port-Knocking* hace que sea muy sencillo el inutilizar dicho mecanismo de seguridad). Un ejemplo de cómo un ataque de este tipo afecta a un mecanismo de seguridad puede verse en la Tabla 1.3, donde un sistema con el mecanismo de autenticación es atacado enviando 60.000 intentos de conexión a puertos aleatorios[1].

[1] La descripción completa del ataque puede encontrarse en [79]

Tabla 1.1: Recursos del sistema que ejecuta el mecanismo de seguridad Port-Knocking, con un intervalo de tiempo entre cada muestra de datos de 5 segundos. Las filas en negrita representan los momentos en los que el ataque tuvo lugar. Las últimas dos filas de la tabla muestran cómo aún después de finalizar el ataque, el sistema aún se encontraba procesando los intentos de conexión recibidos

Memoria				CPU		
Swap	Libre	Buffer	Cache	Usuario	Sistema	Libre
0	7536	8920	119524	0	0	100
0	7536	8920	119524	2	0	98
0	7536	8920	119524	0	1	99
0	7536	8920	119524	0	1	99
0	3784	8924	119524	0	0	100
0	**3976**	**8924**	**117472**	**12**	**47**	**42**
0	**3264**	**8916**	**116648**	**23**	**77**	**0**
0	**3628**	**8920**	**112636**	**58**	**42**	**0**
0	**3708**	**8920**	**111668**	**81**	**19**	**0**
0	**3816**	**8920**	**110696**	**85**	**15**	**0**
0	**3960**	**8920**	**109720**	**92**	**8**	**0**
0	**3052**	**8920**	**109732**	**88**	**12**	**0**
0	**3576**	**8928**	**107916**	**50**	**50**	**0**
0	**3832**	**8928**	**106092**	**41**	**59**	**0**
0	**3908**	**8924**	**102180**	**89**	**11**	**0**
0	**3900**	**8928**	**102180**	**83**	**17**	**0**
0	**3900**	**8928**	**102180**	**89**	**11**	**0**
0	**3900**	**8928**	**102180**	**78**	**13**	**9**
0	3896	8928	102180	0	0	100
0	3896	8928	102180	0	0	100
0	7688	8932	102180	1	0	99
0	7688	8932	102180	1	1	98

Desde este punto de vista, es necesario disponer de mecanismos que nos proporcionen información acerca de las características de rendimiento de un nuevo dispositivo de seguridad para así poder dimensionar y conocer las características de nuestro sistema con mayor exactitud.

Es en este punto en el que la importante relación entre conformidad con el estándar y el rendimiento de nuestra solución de seguridad se pone de manifiesto, ya que el uso de técnicas y mecanismos no recogidos en el estándar que define los protocolos de seguridad (y que de hecho, convierten nuestra implementación en incompatible con el mismo), puede llevarnos a una situación en la que, de todas las posibles opciones de algoritmos criptográficos, métodos de autenticación, etc. . . . ofertadas por nuestra implementación, las únicas que pueden ser utilizadas al conectarnos a otras implementaciones son aquellas que ofrecen un peor rendimiento, lo que nos puede conducir a una auto-denegación de servicio.

Esta situación es un problema para los sistemas que ofrecen servicios en red, ya que su capacidad de crecimiento puede verse repentinamente frenada al no disponer de sistemas capaces de dar servicio a todos los clientes que lo necesitan. Pero para los clientes es también un problema, ya que pueden disponer de dispositivos con escasos recursos computacionales (por ejemplo, los teléfonos móviles o las agendas electrónicas) o de ordenadores personales potentes, pero con un elevado número de aplicaciones ejecutándose en ellos, con lo que si uno de los procesos consume demasiados recursos, el resto de aplicaciones se verán afectadas, incidiendo este hecho seriamente en la usabilidad del sistema ([105]).

Adicionalmente, existe el problema de que la medición de los recursos necesarios para realizar cada operación criptográfica de forma aislada en nuestro sistema no es información suficiente, ya que la existencia de comunicaciones seguras establecidas anteriormente afecta al comportamiento de nuestro sistema, siendo diferente el impacto si esas comunicaciones sólo están establecidas (pero no se transmite tráfico) o si se están transmitiendo datos (variando también los resultados en función del volumen de datos que se envíe). Esta información de rendimiento no suele proporcionarse con los conjuntos de pruebas de rendimiento tradicionales, aunque al utilizar protocolos de seguridad la importancia de este factor sea enorme.

1.4. Objetivos de la tesis

Tras analizar todos los motivos expuestos en este capítulo, y siguiendo con las líneas de investigación ya fomentadas en proyectos del NIST estadounidense y por la Unión Europea, con el fin de garantizar y dotar de seguridad las capacidades de interconexión entre las diferentes entidades que hacen uso de las Tecnologías de la Información y Comunicaciones (como que-

da expresado en las líneas de actuación del VII Programa Marco de la Unión Europea), **se hace necesario disponer de las herramientas necesarias que permitan validar y evaluar los mecanismos de seguridad que se utilizan para la interconexión de redes**.

La presente tesis tiene por objetivo general el proporcionar los métodos y mecanismos necesarias para llevar a cabo una evaluación de las implementaciones de protocolos de seguridad que proporcione información fiable e independiente acerca de:

1. **Conformidad con el estándar**: Se analizarán las especificaciones de los protocolos y arquitecturas de seguridad para identificar aquellos aspectos críticos a la hora de establecer la conformidad o no de una implementación con respecto a lo descrito en el estándar y se definirá una metodología que permita obtener la información relevante en este aspecto.

2. **Interoperabilidad**: A partir de los resultados en las pruebas de conformidad con el estándar, y contando con los resultados obtenidos de la evaluación de la conformidad de otras implementaciones, será posible emitirá un informe acerca de las capacidades y limitaciones en la interoperabilidad de la implementación con otras implementaciones.

3. **Rendimiento de la implementación**: Tras llevar a cabo los pasos indicados en la metodología, se proporcionará al usuario la información acerca del rendimiento que se puede obtener de la implementación analizada, tanto en aspectos tradicionalmente evaluados en conjuntos de pruebas de rendimiento para redes como en otros aspectos no evaluados pero que resultan de interés en el ámbito de los protocolos de seguridad, como ya se ha apuntado anteriormente.

4. **Sugerencias de configuración**: Se definirán las directrices que permitirán, a partir de los resultados anteriores (tanto los relevantes a conformidad como al rendimiento), proporcionar información acerca de las configuraciones que permiten maximizar las posibilidades de interconexión al tiempo que se indican las limitaciones de rendimiento que se presentan con dicha configuración.

Para poder alcanzar las metas propuestas, los objetivos detallados que se marca la presente tesis son **la identificación de los parámetros que influyen en la conformidad con el estándar**, **la identificación de parámetros de rendimiento** que resultan de interés en implementaciones de protocolos de seguridad y el **establecimiento de un marco de análisis y desarrollo para protocolos y arquitecturas de seguridad** a partir del cual poder desarrollar **una metodología que permita validar y evaluar implementaciones de protocolos y arquitecturas de**

seguridad. Por su parte, la **aplicación de la metodología a la arquitectura de seguridad IPsec** junto con el **desarrollo de una plataforma de validación y evaluación remota de implementaciones IPsec** son los objetivos que permitirán evaluar esta tesis. A continuación analizaremos cada uno de estos objetivos con mayor profundidad.

1.4.1. Identificación de parámetros de conformidad con el estándar

En el documento con el que se estandariza un protocolo de seguridad aparecen especificados en lenguaje natural gran cantidad de requisitos, formatos, estructuras de datos, operaciones, suites criptográficas, etc..., que cualquier implementación del protocolo deberá respetar. Sin embargo, el tratar de realizar una validación de una implementación leyendo el estándar del protocolo representa una tarea prácticamente imposible de llevar a cabo, debido tanto a la cantidad de elementos a tener en cuenta como a su diversa naturaleza. Adicionalmente, el hecho de que el estándar se encuentre redactado en lenguaje natural no ayuda a identificar de manera eficaz ni eficiente dichas características que necesitamos.

Por este motivo, el primer objetivo que la presente tesis aborda es la identificación de aquellas características de los protocolos y arquitecturas de seguridad estandarizados que deberán ser tenidas en cuenta al evaluar la conformidad. Con el fin de facilitar su posterior tratamiento y análisis de cara al desarrollo de la metodología, en esta fase se procederá a identificar los parámetros por los que estas diferentes características pueden agruparse. Es decir, todas aquellas características definidas en el estándar que deban ser evaluadas siguiendo técnicas o procedimientos similares, deberán ser agrupadas y presentadas como una "familia" de características, con el fin de que los métodos que se determinen como idóneos para evaluar cada una de esas características pueda aplicarse a todos los demás.

1.4.2. Identificación de parámetros de rendimiento

Como se ha comentado anteriormente, los parámetros de rendimiento que podemos obtener a partir de los conjuntos de pruebas de rendimiento de red actuales pueden no ser los más adecuados para las implementaciones de protocolos de seguridad, bien por problemas de implementación, bien porque la información obtenida no satisface nuestras necesidades como administradores de la red. Como se verá más detalladamente en el Capítulo 4, las operaciones criptográficas que utilizan los protocolos de seguridad hacen difícil predecir los resultados de rendimiento, de forma que no es factible prever a priori cuál va a ser el comportamiento de una determinada implementación de un protocolo de seguridad, ni siquiera en el caso de disponer

de la información que proporciona el fabricante para un dispositivo, equipo o software similar, ya que cualquier cambio en la arquitectura del sistema, en el procesador, en la memoria, etc.... tiene repercusiones en la ejecución de las operaciones criptográficas.

Para afrontar esta problemática, en esta tesis se analizarán las especiales características de los protocolos de seguridad, para así obtener un conjunto de parámetros de rendimiento que es necesario conocer para planificar las capacidades de operación de la implementación analizada. Los parámetros que se identifiquen deberán cubrir las necesidades de información para todo tipo de implementaciones (hardware, software, ...), posibles modos de funcionamiento (pasarela, equipo final), tipos de usuario (usuario personal, administrador de red corporativa), etc....

1.4.3. Establecimiento de un marco de análisis y desarrollo para protocolos y arquitecturas de seguridad

El siguiente de los objetivos planteados dentro de esta tesis doctoral se centra en el desarrollo de un marco de análisis para protocolos y arquitecturas de seguridad, de tal forma que los estudios que se lleven a cabo para identificar los parámetros de conformidad y rendimiento sean extrapolables a cualquier protocolo o arquitectura de seguridad, así como a otras soluciones de seguridad llevando a cabo únicamente modificaciones mínimas en el trabajo realizado.

Asimismo, en este marco se englobarán los desarrollos para la implementación de la plataforma de validación y evaluación remota, de forma que, en la medida de lo posible, las librerías desarrolladas permitan su posterior aplicación a otros protocolos, mecanismos y arquitecturas de seguridad, así como la actualización del juego de pruebas a las que se someten las implementaciones estudiadas.

1.4.4. Metodología de validación y evaluación de implementaciones de protocolos y arquitecturas de seguridad

Una vez conocidos los parámetros que deben ser evaluados, tanto en lo referente a la conformidad con el estándar como a parámetros de rendimiento, en esta tesis se presentará una metodología para la evaluación de implementaciones de protocolos y arquitecturas de seguridad. En esta metodología se propondrán los principios, prácticas y procedimientos para evaluar las implementaciones utilizando para ello las redes de comunicaciones que conecten la implementación que se desea evaluar con el sistema "evaluador".

Esta metodología presentarán, de forma estructurada y razonada, los métodos que permitirán obtener la información necesaria para llevar a cabo la validación de la conformidad y evaluación del rendimiento de implemen-

taciones de protocolos y arquitecturas de seguridad. Adicionalmente, en la metodología también se incluirán las pautas y recomendaciones que deberán tenerse en cuenta a la hora de ofrecer guías de configuración orientativas o comparaciones de los resultados obtenidos.

1.4.5. Aplicación de la metodología a la arquitectura de seguridad IPsec

Como consecuencia directa de la definición de la metodología ya comentada, surge la necesidad de llevar a cabo la aplicación de dicha metodología a la validación y evaluación de implementaciones de un protocolo o arquitectura de seguridad concretas. En esta tesis se ha decidido aplicar la metodología a la arquitectura de seguridad IPsec, por sus características de aceptación e implantación, proyección futura y completitud en cuanto a servicios y mecanismos de seguridad. Esta aplicación de la metodología a una arquitectura de seguridad concreta permitirá obtener una referencia tanto del paso de la metodología genérica a conjuntos de pruebas concretos para un protocolo de seguridad determinado, como de soluciones y propuestas para solventar aquellas dificultades que aparezcan en dicha transición.

Esta aplicación de la metodología deberá contemplar todas las fases, procedimientos, mecanismos y métodos de los que consta la metodología, pero siempre teniendo en cuenta cuáles son las características de la arquitectura a la que se van a aplicar dichos mecanismos.

1.4.6. Desarrollo de una plataforma de evaluación remota de implementaciones de IPsec

Finalmente se desarrollará de una plataforma de validación y evaluación remota de implementaciones de IPsec que desarrolle todo lo especificado por la aplicación de la metodología a la arquitectura de seguridad IPsec. La plataforma resultante deberá cubrir todos los aspectos desarrollados en la aplicación, desde el análisis de los aspectos que limitan la conformidad con el estándar, hasta la generación de esquemas de configuración. Esta implementación deberá servir como modelo para posteriores optimizaciones o mejoras, al tiempo que propondrá soluciones a problemas de implementación que se presenten.

La implementación de la plataforma conlleva un doble desarrollo: por un lado es necesario generar todas las pruebas individuales que evalúan una única característica descrita en la la aplicación de la metodología, y por otro es necesario después integrar todas esas pruebas de concepto en una plataforma única, con un interfaz desde el que el usuario pueda interactuar. Adicionalmente, la plataforma que se desarrolle en esta tesis doctoral deberá tener en cuenta la posibilidad de integrarse con otras herramientas.

1.5. Organización de la tesis

Esta tesis doctoral se organiza de la siguiente manera: En el capítulo 2 se revisará el estado de la cuestión en los aspectos de estandarización y validación de la seguridad, analizando las metodologías de validación de la seguridad del software y la validación de una implementación de un protocolo de seguridad (sección 2.2). En este apartado se estudiarán los diferentes enfoques de la evaluación de protocolos de seguridad en la actualidad, utilizando para ello un proyecto representativo de cada uno de estos enfoques (apartado 2.2.4) para posteriormente revisar las características de diferentes conjuntos de pruebas de rendimiento para dispositivos de red (sección 2.3).

En el capítulo 3 se afrontará el primero de los objetivos marcados para esta esta tesis: el análisis de las características de conformidad con los estándares de IPsec, prosiguiendo en el capítulo 4 con análisis de los parámetros de rendimiento que es necesario evaluar en una implementación de IPsec. Como resultado de estos análisis, en el capítulo 5 se presentará la metodología de evaluación de implementaciones de protocolos y arquitecturas de seguridad que recoge las aportaciones de los capítulos anteriores y las convierte en el capítulo 6 en una guía para evaluar las implementaciones de la arquitectura de seguridad IPsec. Finalmente, en el capítulo 7 se estudiará la implementación de esta metodología propuesta, para llevar a cabo la validación y evaluación remota de implementación de IPsec.

El capítulo 8 se dedica por completo a presentar los resultados de evaluar implementaciones comerciales de IPsec utilizando la plataforma desarrollada, analizando los resultados obtenidos.

Por último, en el capítulo 9 se presentarán las conclusiones de esta tesis doctoral, tanto en lo referente a las aportaciones realizadas en la misma (comparándolas con los objetivos propuestos en dicha tesis) como a los documentos y desarrollos que surgen de la misma (sección 9.1).

Capítulo 2

Estado de la Cuestión

En este capítulo analizaremos cuál es el estado de la cuestión tanto en la estandarización y evaluación de la seguridad, como en el análisis de rendimiento, especialmente de los protocolos y arquitecturas de seguridad. En primer lugar se llevará a cabo un estudio de la situación de los estándares utilizados en el ámbito de la protección de las comunicaciones.

Por lo tanto, el capítulo comenzará describiendo en la sección 2.1 el estado actual de los protocolos y arquitecturas de seguridad que han sido estandarizados, así como los estándares más relevantes desde el punto de vista de la seguridad en las comunicaciones en cuanto a mecanismos criptográficas. A continuación, en la en la sección 2.2 pasaremos a analizar el estado de la cuestión en el ámbito de la validación de la seguridad, donde se estudiarán los diferentes enfoques en la validación de la seguridad que es posible encontrar en la literatura científica en la actualidad. En esta sección merecen especial atención los apartados sobre metodologías de evaluación de seguridad (apartado 2.2.5) y sobre validación de protocolos de seguridad (apartado 2.2.4), en los que se analizan los temas que tienen una mayor relación con el ámbito de esta tesis. Finalmente, en la sección 2.3 se analizará el estado de la cuestión en lo tocante al análisis de rendimiento de protocolos de red, haciendo especial hincapié en los análisis de rendimiento centrados en los protocolos de seguridad (apartado 2.3.1).

2.1. Estandarización de la seguridad

En esta sección se presentarán las contribuciones de la comunidad científica en los pasados años al campo de la estandarización de la seguridad, especialmente en cuanto a los protocolos y arquitecturas de seguridad se refiere. Adicionalmente, también se revisarán los trabajos de estandarización de herramientas criptográficas más relevantes para los protocolos de seguridad.

2.1.1. Protocolos y arquitecturas de seguridad

En este apartado se analizarán los protocolos y arquitecturas de seguridad en uso actualmente, de forma que podamos conocer cuál ha sido su evolución, los servicios de seguridad que ofrecen y los mecanismos y herramientas criptográficas que utilizan para proteger la información, así como el nivel de la pila de comunicaciones al que trabaja cada uno de los protocolos o arquitecturas estudiados. El análisis se llevará a cabo sobre el protocolo TLS y la arquitectura de seguridad IPsec, debido a la amplia aceptación de ambos por parte de la comunidad científica, y su amplia (casi totalitaria en algunos casos) adopción de ambos para la protección de los servicios de Internet. La aceptación de ambos ha sido tal, que la aparición de nuevos dispositivos y sistemas de comunicación que no podían implementar estos protocolos (bien por los requisitos computacionales necesarios, bien por problemas al implantar la pila de protocolos requerida), ha llevado aparejado el desarrollo de nuevos protocolos que sustituyesen a TLS o IPsec. El caso más notorio es el de WTLS (*Wireless Transport Layer Security*, [57]), que ofrece funcionalidad similar a TLS a los dispositivos que operan con WAP (*Wireless Application Protocol*, [56]), principalmente teléfonos móviles.

2.1.1.1. SSL y TLS

Descripción

El protocolo SSL (*Secure Socket Layer*) fue desarrollado por la compañía Netscape Communications desde el inicio del desarrollo del navegador web de dicha compañía. Desde los trabajos en el navegador Mosaic por la *National Center for Supercomputing Applications* (NCSA) en 1.993 ([106]) la necesidad de incluir algún mecanismo de seguridad se hizo evidente. Por este motivo, y de cara a incluirse con la versión 1.0 de Netscape Navigator, la compañía desarrolladora de este navegador inició el diseño de la versión 1.0 de SSL, proceso que se culminaría sólo 8 meses más tarde, a mediados de 1.994. Posteriormente, a finales de ese mismo año Netscape distribuía la primera versión de su navegador con soporte para la versión 2.0 del protocolo SSL. Esta nueva versión estaba orientada a subsanar deficiencias que se detectaron en la versión 1.0 del protocolo, principalmente en la gestión de claves criptográficas.

La adopción de SSL como un estándar de facto por parte de la industria se afianzaría en 1.995, cuando Microsoft Corporation lanzó su navegador Internet Explorer e incluyó en el mismo el soporte para SSL, que, recordemos, en aquel momento seguía siendo un protocolo desarrollado por una compañía rival y cuyo proceso de estandarización no había comenzado. Por este mismo motivo, Microsoft publicó en 1.995 la especificación de un nuevo protocolo que sustituiría a SSL: el protocolo PCT (*Private Communication*

Technology) versión 1.0. Aunque este intento de Microsoft por hacerse con el control de los protocolos de seguridad en Internet no fructificó, las propuestas y mejoras que se recogían en esa especificación fueron incorporadas a SSL en su versión 3.0, publicada por Netscape a finales de ese mismo año.

En este punto de la historia tuvo lugar un cambio en el diseño y desarrollo de SSL: hasta el momento, Netscape Corporation había desarrollado las tres primeras versiones del protocolo siguiendo un proceso abierto y en el que animaba a participar a otras compañías y comunidades del sector. Sin embargo, el que la compañía siguiera siendo propietaria del protocolo (de hecho, Netscape Corporation solicitó con éxito una patente sobre este protocolo) no animaba a potenciales competidores a colaborar, al tiempo que la comunidad internacional miraba con recelo las posibles implicaciones futuras de este modelo de desarrollo. Por este motivo, en Mayo de 1.996 Netscape cedió el desarrollo de SSL, que pasó a encontrarse bajo el control del *Internet Engineering Task Force*, que es el responsable del desarrollo del mismo en la actualidad. Debido a la propiedad de Netscape del nombre SSL, una de las primeras acciones del IETF fue renombrar el protocolo a TLS (*Transport Layer Security*). La versión 1.0 de este protocolo (ya estandarizado) se publicó en enero de 1.999 ([48]), y está basada en SSL 3.0 (hasta el punto de que las diferencias entre SSL 3.0 y TLS 1.0 son menores que las existentes entre SSL 2.0 y SSL 3.0).

En la actualidad la gran mayoría del software que hace uso de Internet de una forma u otra cuenta con soporte para SSL o TLS, ya que este protocolo se ha convertido en el estándar de protección de información en Internet. Dadas las similitudes entre ambos protocolos, de aquí en adelante hablaremos de TLS para referirnos indistintamente a ambos protocolos.

Funcionamiento

TLS es un protocolo de seguridad que se sitúa sobre la capa de transporte de la pila de comunicaciones, más concretamente, sobre el protocolo TCP. Una de sus características principales es que al introducirse en la pila de protocolos, lo hace de forma transparente a las capas inmediatamente superior e inferior, de forma que no es necesaria ninguna modificación al protocolo TCP (ya que TLS utilizará los mecanismos de comunicación que ofrece TCP, en concreto, los sockets) ni a la capa de aplicación (ya que ésta podrá hacer uso de los sockets seguros de TLS prácticamente de igual forma que utilizaba los sockets TCP). Un esquema de este funcionamiento puede verse en la Figura 2.1. Gracias a este funcionamiento, TLS crea una capa adicional en la pila de comunicaciones que permite a las aplicaciones que securicen sus comunicaciones sin tener que realizar modificaciones importantes en su código.

En cuanto a los mecanismos internos que utiliza TLS para proteger

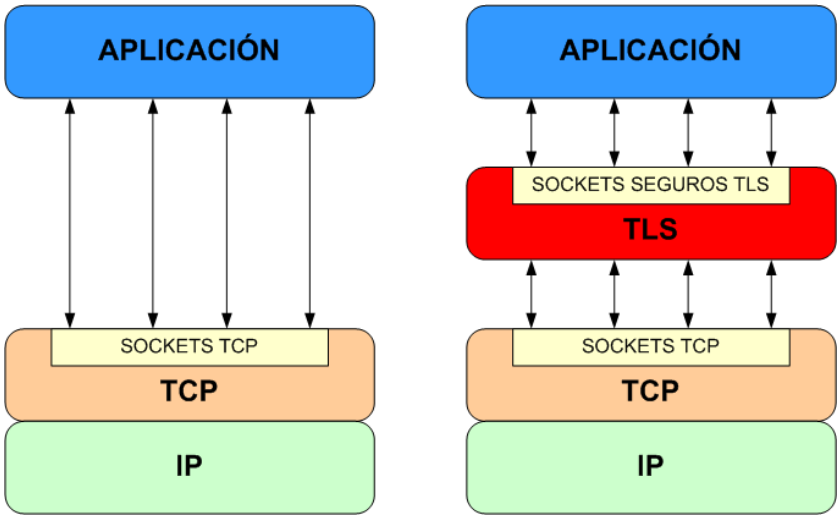

Figura 2.1: Pila de protocolos TCP/IP tradicional (izquierda) y pila TCP/IP tras añadir la capa de TLS

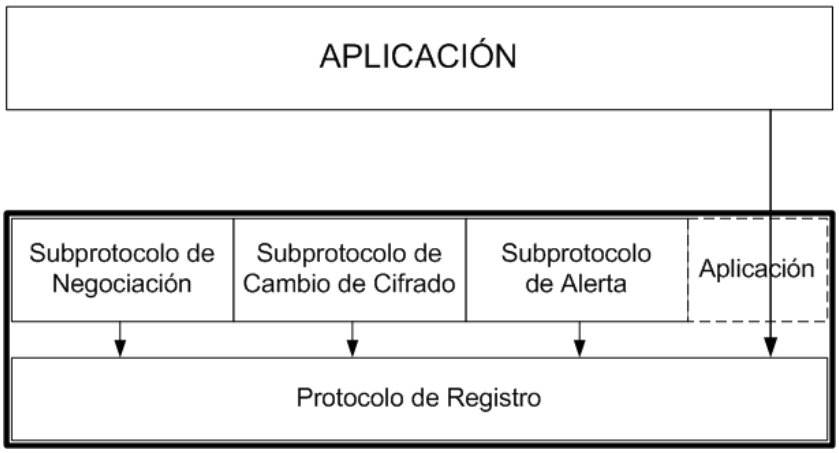

Figura 2.2: Pila de protocolos TLS. La capa de aplicación y los subprotocolos de negociación, alerta y cambio de cifrado generan paquetes que son protegidos y enviados por el protocolo de registro

la información, TLS está compuesto por varios subprotocolos específicos para llevar a cabo tareas concretas. La pila de subprotocolos de TLS puede verse en la Figura 2.2. La función de cada uno de estos subprotocolos es la siguiente:

- **Protocolos de Negociación** *(Handshake Protocols)*: El protocolo de negociación de TLS es el encargado de permitir a los diferentes extremos de la comunicación negociar los parámetros de seguridad que se aplicarán posteriormente para proteger la información (el protocolo de registro será el encargado de aplicar los parámetros que se negocien con este protocolo), así como de autenticar a las entidades que toman parte en la comunicación, instanciar los parámetros de seguridad negociados y notificar las posibles condiciones de error que se pudieran dar. Cada una de estas funciones es llevada a cabo por un subprotocolo diferente, por lo que el protocolo de negociación realmente consta de 3 subprotocolos, que son los siguientes:

  - **Subprotocolo de Cambio de Cifrado** *(Change Cipher Spec Protocol)*: Este subprotocolo es el encargado de notificar al otro extremo de la comunicación segura un cambio en los parámetros utilizados para proteger la información, mediante el envío de un único mensaje (securizado con los parámetros de protección "antiguos"), que marca el fin del uso de los parámetros anteriores, ya que todos los mensajes posteriores se protegerán con los nuevos parámetros.
  - **Subprotocolo de Negociación** *(Handshake Protocol)*: El subprotocolo de negociación es el encargado de negociar los parámetros que regirán la seguridad de la comunicación. Los parámetros más importantes que se negocian son: la versión del protocolo a utilizar (pudiéndose utilizar cualquier versión de SSL o TLS que soporten las partes implicadas en la comunicación), los algoritmos de cifrado, firma digital y compresión que se utilizarán, los parámetros de autenticación y las técnicas de clave pública con las que se generarán los secretos compartidos. Los pasos concretos que se han de llevar a cabo para finalizar con éxito este proceso son:
    - Intercambiar los mensajes iniciales (*Hello Messages*) para acordar los algoritmos a utilizar
    - Intercambiar valores aleatorios con los que generar material criptográfico
    - Comprobar si es posible continuar una sesión SSL / TLS anterior
    - Intercambiar la información criptográfica necesaria para que

ambas partes puedan disponer de un secreto compartido (**secreto pre-maestro**).

- Intercambiar la información criptográfica necesaria para que ambas partes puedan autenticarse (el servidor debe hacerlo de forma obligatoria; para el cliente es opcional).
- Generar un secreto compartido (**secreto maestro**) a partir del secreto pre-maestro y los valores aleatorios anteriores.
- Proporcionar los parámetros y valores de seguridad necesarios al protocolo de registro.
- Permitir a la otra entidad verificar que los parámetros de seguridad y valores criptográficos (especialmente el secreto maestro) se han intercambiado sin incidencias, y sin que haya evidencia de ningún ataque durante esta fase de negociación.

Una vez que estos parámetros han sido acordados, el subprotocolo de negociación se encargará de proporcionar al protocolo de registro la información necesaria para que pueda proceder a proteger la información.

- **Subprotocolo de Alerta** *(Alert Protocol)*: El subprotocolo de alerta es el encargado de notificar al resto de protocolos y subprotocolos, así como al otro extremo de la comunicación, de una condición de error, informando al tiempo de la gravedad de dicha condición. En caso necesario, un mensaje de alerta puede llegar a implicar el fin de la comunicación.

- **Protocolo de Registro** *(Record Protocol)*: Es el encargado de tomar los mensajes que las capas superiores de TLS o la capa de aplicación de TCP desean enviar, y fragmentarlos en bloques del tamaño adecuado, comprimirlos, aplicarles los mecanismos de confidencialidad y autenticidad necesarios y transmitirlos a las capas inferiores de la pila de comunicaciones. A la hora de recibir los mensajes, este protocolo será el encargado de realizar las operaciones inversas: comprobar la autenticidad del mensaje, descifrarlo, descomprimirlo, reensamblarlo y transmitirlo a la aplicación o subprotocolo de TLS al que vaya dirigido ese mensaje.

Un aspecto importante del funcionamiento de TLS es la diferencia entre los conceptos de *sesión* y *conexión*. Una sesión es el conjunto de parámetros de seguridad y valores utilizados para proteger la comunicación entre dos sistemas que utilizan TLS, mientras que una conexión es un flujo de datos entre esos dos sistemas, que se protege utilizando los parámetros y valores de la sesión TLS en la que se instancia dicha conexión. Los parámetros que definen una sesión son:

- Un identificador unívoco, arbitrario y elegido por el servidor.

- Un certificado del sistema con el que se establece el canal seguro.
- Un método de compresión que aplicar a la información antes de ser protegida.
- Una suite criptográfica con la que proteger la información
- Un secreto maestro de 48 bits, compartido entre el cliente y el servidor.
- Un indicador de si la sesión es reiniciable o no, es decir, de si los valores y parámetros negociados deben ser almacenados para usos posteriores sin necesidad de tener que llevar a cabo otra negociación.

Las suites criptográficas definen conjuntos de algoritmos y mecanismos de seguridad que se utilizarán para proporcionar los servicios de seguridad deseados. Por ejemplo, una suite criptográfica aceptable en la versión 1.1 de TLS es `TLS\_DHE\_DSS\_WITH\_DES\_CBC\_SHA`, que especifica los siguientes parámetros:

- Diffie-Hellman efímero con firmas DSS ([112]) como algoritmo de intercambio de claves.
- El cifrador de bloque DES operando en modo CBC para cifrar la información.
- SHA como función resumen.

Servicios de seguridad que proporciona

TLS proporciona los servicios de seguridad de **confidencialidad** e **integridad**, y, opcionalmente, **autenticación**. Dado que TLS soporta tres niveles de autenticación (mutua autenticación, autenticación del servidor y ausencia de autenticación), el servicio de autenticación puede estar presente o no. Cuando existe autenticación de al menos una de las partes, el túnel criptográfico es resistente a ataques de hombre en el medio, ya que para autenticarse cada entidad debe presentar una cadena de certificados que conduzca a una autoridad de certificación aceptada por la otra parte, utilizando para ello los servicios de una infraestructura de clave pública. Mediante el uso de las suites criptográficas que se negocian, cada sistema involucrado es capaz de proteger la información que envía, al tiempo que la inclusión de códigos de autenticación del mensaje (MAC, *Message Authentication Codes* [137]) permite detectar modificaciones ilegítimas a la información que fluye por el túnel criptográfico.

2.1.1.2. IPsec

Descripción

La arquitectura de seguridad IP (IPsec) es una propuesta del Internet Engineering Task Force (IETF) para dotar de protección basada en criptografía con alto nivel de seguridad a la capa de red IP (tanto para la versión 4 (IPv4) como para la versión 6 (IPv6)), manteniendo la interoperatividad entre los dispositivos o equipos que implementen esta arquitectura de seguridad. Como arquitectura de seguridad, está compuesta por múltiples protocolos internos, algunos de los cuales son obligatorios para la correcta protección de la información, mientras que otros son meras propuestas.

Entre los objetivos de IPsec se encuentra el que, cuando se encuentra correctamente implementado y desplegado, no debe afectar a las aplicaciones ni a los usuarios que no protejan sus comunicaciones con esta arquitectura. Además, la modularidad de los protocolos que lo conforman (todos ellos son criptográficamente independientes) permite que las aplicaciones y usuarios que sí hacen uso de la arquitectura puedan recibir una protección acorde a sus necesidades, y que afecte lo menos posible a las operaciones que se llevan a cabo.

La especificación de IPsec se hizo pública en 1.995, con la publicación de las RFCs en las que se definían las especificaciones para una arquitectura de seguridad a nivel de red ([11], [9], [10], [103], [83]), aunque en 1.998 una revisión general de la arquitectura dejó obsoletas las antiguas especificaciones, liberándose una nueva versión de las mismas en las que se estandariza una arquitectura conceptualmente similar a la primera versión, pero incompatible con la misma ([90], [88], [89], [101], [85], [54], [70]). A finales de 2.005 se publicó una nueva versión de las especificaciones, en la que se reorganizaron los protocolos internos, así como algunos de los servicios de seguridad ofrecidos ([91], [86], [87], [136]).

IPsec consta de varios protocolos, cada uno de los cuales se encarga de llevar a cabo una tarea determinada dentro de la arquitectura IPsec: gestión de claves criptográficas, autenticación, confidencialidad, ...; dependiendo del conjunto de protocolos que se utilice para proteger una determinada comunicación, los servicios de seguridad que se ofrecerán variarán. Algunos de estos protocolos son de obligada implementación, mientras que otros son propuestas del IETF que sirven de ejemplo para el desarrollo de posibles protocolos alternativos (aunque posteriormente estas propuestas hayan sido aceptadas como estándares de facto).

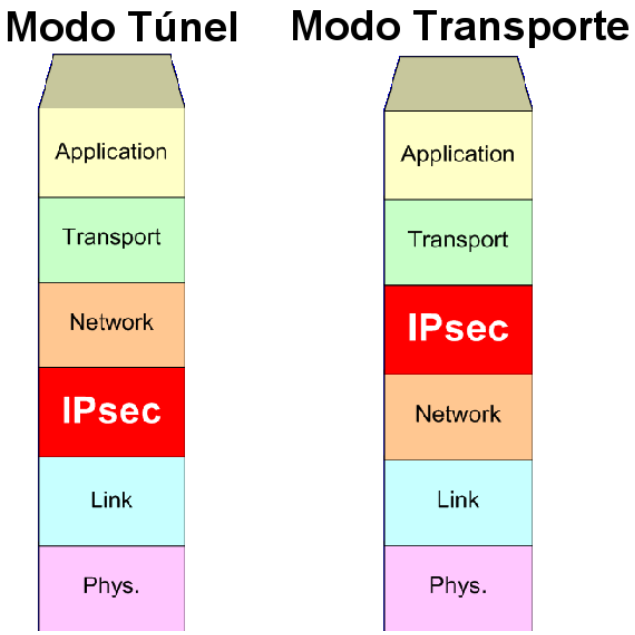

Figura 2.3: Pila de protocolos IPsec en modo transporte y en modo túnel

Funcionamiento

IPsec tiene dos modos de funcionamiento independientes, con objetivos diferentes. Por un lado se define el *modo transporte* para los datos, en el que se protegen los datos de nivel superior a IP (TCP, ICMP, etc.); en modo *modo túnel*, sin embargo, lo que se protegen son los propios paquetes de la capa IP, encapsulándolos en otro paquete IP. Un esquema de la pila de protocolos resultante en ambos casos puede verse en la Figura 2.3, donde se muestra qué información se protege en cada caso.

Por otro lado, IPsec está compuesto varios protocolos que se encargan de las tareas de gestión de información criptográfica y protección de la información. La gestión de las claves recae sobre los protocolos IKE e ISAKMP en la especificación de 1.998 y sobre IKE en la especificación de 2.005 (o sobre cualquier protocolo que realice sus funciones, ya que los mecanismos de gestión de claves propuestos en los estándares son recomendaciones, y pueden ser sustituidos por cualquier otra solución que cumpla los requisitos). Por su lado, la protección de la información es responsabilidad de los protocolos ESP y/o AH, dependiendo de los parámetros de seguridad que se hayan negociado. Por lo tanto, y teniendo en cuenta que la negociación de los parámetros de seguridad se lleva a cabo en dos fases que se explicarán a continuación, el papel que juega cada protocolo en IPsec se ve reflejado en la Figura 2.4.

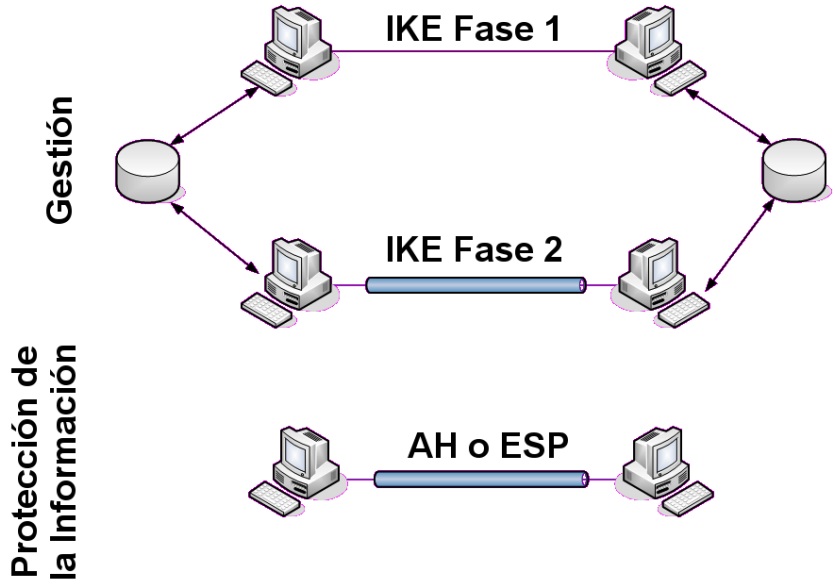

Figura 2.4: Papel que juega cada protocolo de IPsec en el establecimiento y protección de túneles seguros

En la fase de negociación de IPsec el objetivo es acordar los parámetros criptográficos que regirán el posterior intercambio de información. Sin embargo, en lugar de llevar a cabo una negociación similar a la de TLS, en IPsec el proceso se lleva a cabo de tal forma que el proceso de autenticación sólo se requiere una vez entre cada pareja de dispositivos IPsec, de tal forma que negociaciones posteriores no deberán volver a llevar a cabo ese mecanismo. De forma breve, el proceso de negociación de claves se lleva a cabo de la siguiente manera:

- En la **Fase 1** de la negociación los sistemas que establecerán el túnel seguro se autentican mutuamente utilizando cualquiera de los métodos previstos para ello en el estándar, y acuerdan los parámetros de seguridad que se utilizarán en la Fase 2 de la negociación.
- La **Fase 2** de la negociación se lleva a cabo cada vez que es necesario establecer una nueva asociación de seguridad entre dos entidades IPsec. En esta fase (en la que todo el tráfico ya se transmite protegido por los parámetros negociados en la Fase 1), se negocian los parámetros de seguridad concretos con los que se protegerá la comunicación de capas superiores. Entre los aspectos que se negocian se encuentran el protocolo a utilizar para proteger la información: ESP o AH.

Como podemos ver, de esta forma es posible negociar múltiples túneles seguros llevando a cabo una única autenticación de las partes, lo que reduce

considerablemente la cantidad de proceso a llevar a cabo, a costa de una mayor complejidad en la negociación de claves. Adicionalmente, con esta estructura se protege a cada una de las fases de potenciales problemas de seguridad en capas anteriores: si un atacante consiguiera hacerse con las claves negociadas en una Fase 1, los parámetros de seguridad negociados en ejecuciones de la Fase 2 anteriores no se comprometen, por lo que esos túneles seguros pueden mantenerse operacionales.[1].

Otro concepto fundamental que ya ha sido mencionado son las asociaciones de seguridad (*Security Associations*, SA), que son conjuntos de servicios de seguridad que se aplicarán a las comunicaciones que cumplan con un patrón determinado. Las asociaciones de seguridad constan de un identificador único (*Security Parameter Index*, SPI), los algoritmos de cifrado y autenticación a utilizar (en caso necesario), el tiempo de vida de la asociación de seguridad, el modo de funcionamiento, y el protocolo que se utilizará para proteger la información. Opcionalmente también puede incluir el contenido de la ventana de datos para detección de ataques de reenvío de paquetes, el valor de la MTU y contadores para los números de secuencia de AH o ESP. Las definiciones de las asociaciones de seguridad se almacenan en la base de datos de políticas de IPsec (*Security Policy Database*, SPD), en la que también se incluyen los parámetros del tráfico que se protegerá con cada una de las asociaciones de seguridad.

Por último, cabe destacar que una implementación de IPsec puede operar como un equipo intermedio que proporciona el servicio de seguridad a otros equipos (lo que se suele llamar un *IPsec gateway* o *pasarela IPsec*), o como un dispositivo o equipo independiente que protege su tráfico con IPsec, o cualquier combinación de este tipo de modos de operación, como se muestra en la Figura 2.5. La protección que ofrecerá cualquiera de estas implementaciones de IPsec serán las acordes al conjunto de políticas de seguridad definidas en el dispositivo por el administrador. Para determinar cuál de estas políticas es la que debe utilizarse para cada canal de comunicación o paquete de datos concreto, IPsec establece mecanismos, llamados *Selectors*, que permiten realizar esta discriminación basándose en datos de la cabecera IP o de la cabecera del siguiente nivel.

Servicios de seguridad que proporciona

Los servicios de seguridad que ofrece la arquitectura de seguridad IPsec dependen en gran medida de qué protocolos son los utilizados para proteger la información que se desea transmitir. Por lo tanto, es necesario realizar

[1] Para que esta afirmación sea totalmente cierta es necesario que se utilice la opción de *Perfect Forward Secrecy* (PFS), que independiza totalmente las claves de cada fase de las utilizadas en fases anteriores

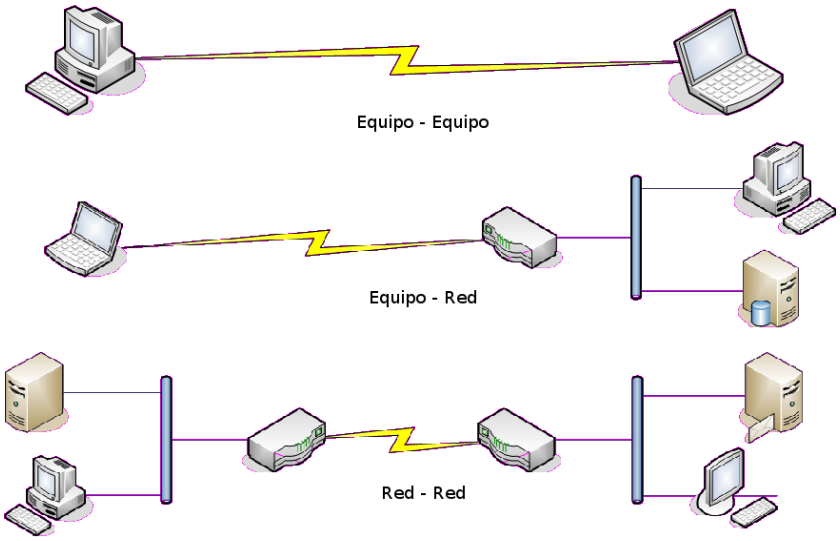

Figura 2.5: Posibles modos de funcionamiento de IPsec, desde el punto de vista de su operación como pasarela o como dispositivo final

un estudio de los servicios de seguridad que proporciona cada uno de estos protocolos para conocer cuál es el conjunto de servicios que se encuentran disponibles en la arquitectura:

- **Authentication Header** (AH): este protocolo proporciona los servicios de *integridad* de la comunicación (siempre teniendo en cuenta que las labores de control de flujo y de la conexión se llevan a cabo por la capa de transporte de la pila de comunicaciones), *autenticación del origen de los mensajes*, y, opcionalmente, *protección contra el reenvío de mensajes*, que aunque es de obligada implementación para el emisor, tiene carácter opcional para el receptor.
- **Encapsulating Security Payload** (ESP): al igual que AH, ESP es capaz de proteger la información transmitida por los túneles criptográficos proporcionando los servicios de *integridad*, *autenticación del origen de los mensajes* y *protección contra el reenvío de mensajes*. Adicionalmente, con ESP es posible contar con los servicios de *confidencialidad*, ya que la información se transmite cifrada entre emisor y receptor, y *protección contra el análisis de tráfico*

En cuanto a los servicios de seguridad, ESP y AH ofrecen una cobertura diferente de los mismos, como pasamos a detallar:

- AH, cuyo soporte en una implementación IPsec es *opcional*, ofrece servicios de integridad y autenticación del origen de los datos, con la posibilidad opcional (a elección del receptor) de utilizar técnicas para

evitar el reenvío de paquetes. Adicionalmente, proporciona servicios de control de acceso mediante la distribución de claves criptográficas según se defina en las políticas de seguridad que gobiernan IPsec.

- ESP, cuyo soporte en una implementación IPsec es *obligatorio*, ofrece los mismos servicios que AH, y además proporciona confidencialidad. Esta confidencialidad también se aplica parcialmente a los datos relativos al tráfico original (como el tamaño de los datos, etc.). Adicionalmente, ESP es capaz de ofrecer resistencia a ataques de reenvío de mensajes.

Adicionalmente, es posible contar con protección frente a posibles ataques de análisis de tráfico si la arquitectura de seguridad IPsec se utiliza para funcionar en modo túnel, de forma que la información que un posible atacante pueda recuperar se limite a la mínima indispensable para encaminar los mensajes desde el equipo o red de origen hasta el destinatario.

2.1.2. Mecanismos criptográficas

La estandarización de mecanismos criptográficos se ha llevado a cabo por diferentes organismos de estandarización, dependiendo del alcance y ámbito que se pretendiese abarcar con dicho estándar. Por un lado, organismos como IEEE e ISO han estandarizado en los últimos años mecanismos de seguridad integrados en protocolos (como por ejemplo, los mecanismos de protección WEP y WPA definidos en el estándar 802.11b, o los mecanismos de seguridad incluidos en el estándar 802.11i ([43] y [44] respectivamente)) o arquitecturas de seguridad,(como las descritas en el estándar ISO/IEC 10181 ([42]), en el que se revisan los marcos de trabajo para la seguridad en sistemas abiertos).

Por otro lado, en estos procesos de estandarización de herramientas criptográficas también han participado organismos como el NIST estadounidense (con repercusión legal únicamente en Estados Unidos, pero que crea estándares de facto en toda la comunidad internacional), la Unión Internacional de Telecomunicaciones (*Intenational Telecommunications Union*, ITU) y el Internet Engineering Task Force, que en su proceso de estandarización de las herramientas utilizadas en el desarrollo y utilización de Internet también estandariza las suites criptográficas válidas para su uso por los diferentes protocolos y arquitecturas de seguridad (como por ejemplo al estandarizar AES_CMAC ([142])). Sin embargo, este organismo también es el responsable de la estandarización de las funciones resumen MD5 ([134]) y los algoritmos de cifrado CAST ([2] y [3]). Por su parte, el ITU (a través de su rama de estandarización, la ITU-T) ha liberado estándares tan interesantes y extendidos como el estándar de clave pública y certificado de atributos, X.509 ([72]), en la que se sientan las bases de las infraestructuras

de clave pública, al especificar el formato de los certificados de clave pública más extendido en la actualidad, y un algoritmo para la validación de un certificado de clave pública siguiendo la jerarquía de la infraestructura de clave pública.

En cuanto a la estandarización de las herramientas de seguridad que lleva a cabo el NIST, este proceso se encuadra en las labores de esta organización, como gestora de los Estándares Federales de Procesamiento de la Información (*Federal Information Processing Standards*, FIPS). Entre estos documentos podemos encontrar las especificaciones de los requisitos para los módulos criptográficos ([111] y [114]), las especificaciones de generadores automáticos de claves y contraseñas ([110]), las especificaciones del algoritmo estándar de cifrado, AES ([113]) y las de la función resumen SHA ([116]).

Finalmente, conviene destacar la existencia de gran cantidad de herramientas criptográficas desarrolladas por la comunidad científica, y que no han sido sometidas a procesos de estandarización por ningún organismo, como es el caso de RIPEMD y RIPEMD-160 ([50]) o el algoritmo de cifrado Blowfish ([138]), que pese a estar ampliamente aceptado y soportado por los sistemas de seguridad, al no estar respaldado por ninguna normativa ni organismo internacional ve cómo su uso queda prácticamente relegado al entorno académico.

2.1.3. Otras estandarizaciones

Además de los ya mencionados procesos de estandarización de protocolos y arquitecturas de seguridad por un lado, y de herramientas criptográficas por otro, en los últimos años se han producido otras estandarizaciones por parte de organismos de estandarización, dirigidos principalmente a entornos industriales y de negocios, en lugar de a los ámbitos científicos en los que encontramos los anteriores estudiados previamente.

Por ejemplo, en esta línea de estandarización nos encontramos con la serie X.800 de la Unión Internacional de Telecomunicaciones, en la que se abordan los problemas de seguridad y se ofrecen marcos de trabajo para la interconexión segura de sistemas abiertos (recomendación X.810, [78]), o modelos de seguridad aplicados a diferentes niveles de la pila de protocolos, estudiando el impacto de diversos factores a los mecanismos de seguridad (recomendaciones X.802 ([77]) y X.803 ([76])).

Estas recomendaciones de la ITU tienen su correspondiente estandarización por parte la Organización Internacional de Estándares (ISO), que ha estandarizado el marco de trabajo para la interconexión segura en la norma ISO/IEC 10181 ([42]), y los modelos de seguridad aplicados a ciertos niveles de la pila de comunicaciones en las normas ISO/IEC 10745 (capas superiores, [40]), ISO/IEC 13594 (capas inferiores, [38]) e ISO/IEC 111577 (capa de red, [39]).

Adicionalmente ISO cuenta con documentos adicionales en los que se estandarizan técnicas y mecanismos de seguridad, siendo la norma mas extendida la ISO/IEC 11770 ([41]), en la que se propone un marco de trabajo y mecanismos de operación para gestionar y operar claves criptográficas, tanto simétricas como asimétricas.

Por último, merece la pena destacar la aportación del ISO al estandarizar las normas ISO/IEC 9646 y 13245 ([37], [36]), ya que en ellas se recogen los aspectos formales de los análisis de conformidad para el análisis de la interoperatividad entre sistemas abiertos, incluyendo una metodología para llevar a cabo dicho análisis. Sin embargo, como se comentará en el apartado 2.2.4 donde se analiza el problema en detalle para los protocolos de seguridad, los análisis de interconexión deben realizarse tanto desde un punto de vista formal del sistema o protocolo en cuestión (como el que proponen las normas) como desde el punto de vista de la implementación del sistema o del protocolo.

2.2. Validación y Seguridad

En esta sección se revisarán las propuestas existentes en la comunidad científica en lo referente a mecanismos de validación relacionados con la seguridad. Como veremos a continuación, por un lado nos encontramos con la validación de la implantación de mecanismos y herramientas de seguridad en sistemas de información, validación llevada a cabo durante el proceso de su desarrollo en el sistema. Tras revisar el estado de la cuestión de estos métodos de validación, analizaremos dos de las más recientes propuestas por parte de organismos de estandarización, el *Information Security in the System Development Life Cycle* y el *Automated Security Functional Testing*, ambos promovidos y desarrollados en el National Institute of Standards and Technology (NIST) estadounidense.

Por otro lado, como un proceso posterior a la fase de desarrollo, más relacionado con el mantenimiento de los sistemas, surgen las guías de configuración de la seguridad, que llevan a cabo un análisis del estado de los mecanismos y servicios de seguridad en el sistema. En esta misma línea surgen los conjuntos de pruebas de seguridad (*security benchmarks* en la literatura inglesa), de los que también estudiaremos las tendencias más representativas.

En cuanto a la validación de protocolos de seguridad, en el apartado 2.2.4 analizaremos el papel de la validación formal de protocolos y su relación con la validación que se propone en esta tesis doctoral. Asimismo, revisaremos las contribuciones de la comunidad científica en cuanto a la validación de implementaciones de esos protocolos, analizando la situación actual de dichas propuestas.

Por último, revisaremos el papel de las metodologías de evaluación de seguridad en el ámbito de las comunicaciones seguras, centrándonos en las propuestas más actuales.

2.2.1. Validación de la seguridad en el desarrollo

En el apartado de validación de una solución de comunicaciones segura, la literatura científica nos proporciona una gran cantidad de material relacionado con dos aspectos fundamentales:

- Validación del desarrollo (normalmente software, aunque también pueden aplicarse a desarrollos hardware)
- Validación y verificación del protocolo de seguridad

En cuanto a la validación del desarrollo, podemos encontrar en la literatura con las metodologías tradicionales de evaluación del software, centradas en asegurar un desarrollo "correcto" en las fases de especificación de requisitos (mediante la inclusión de requisitos de seguridad), diseño, e implementación ([18]). Los ejemplos más conocidos de estas técnicas son los que se encuentran integrados en las propias metodologías de desarrollo y en los métodos formales de desarrollo, como puede verse en [18], [129], [16] y [55].

Estas técnicas tienen por objeto minimizar los errores durante el proceso de desarrollo de tal forma que los resultados se ajusten a los requisitos planteados en las primeras fases del propio desarrollo. Sin embargo, debido a la fuerte dependencia del diseñador en esa especificación de requisitos, es posible que los requisitos se interpreten erróneamente o que sean incompletos, sin que este problema sea detectado por las actuales técnicas de desarrollo seguro. En algunos casos, esta ambigüedad se ha traducido en que las propias medidas de seguridad se vuelvan contra nosotros, como se comenta en [108], donde múltiples casos en los que herramientas de seguridad tales como cortafuegos, detectores de intrusos o cifradores de datos han impedido el desarrollo normal de la actividad que debían proteger. Por lo tanto, es perfectamente posible que un desarrollo hardware o software que haya pasado por las metodologías tradicionales pueda no ser conforme al estándar, y por lo tanto, presentar problemas al establecer una conexión segura con una implementación diferente.

Otras propuestas en esta misma línea que se se aproximan al tema de tesis propuesto son las metodologías de evaluación del software, tales como la *System Security Engineering Capability Maturity Model* (SSE-CMM, [66], [67], [75]), o los *Common Criteria* (CC, [121], [122], [123]), que ha sustituido tanto a *Trusted Computer System Evaluation Criteria* (TCSEC, [119]) como a *Information Technology Security Evaluation Criteria* (ITSEC, [125]).

Todas estas metodologías evalúan de una forma u otra la seguridad de un desarrollo (normalmente software) en un entorno concreto, midiendo el nivel de conformidad con unos requisitos definidos en la metodología en cuestión.

En los últimos años han surgido nuevos enfoques para la validación específica de sistemas y herramientas de seguridad que sí tienen en cuenta los requisitos, bien especificados como estándares o como pruebas formales que el diseño debe superar. Sin embargo, podemos ver que la gran mayoría de la literatura realiza *pruebas de caja blanca* para validar el sistema ([93], [17], [149], [62], [98], [35]). Desgraciadamente, este tipo de pruebas no ayuda a analizar el funcionamiento del sistema tal y como lo ven otros sistemas (es decir, como una *caja negra*), y, dado que estamos hablando de sistemas cuyo objetivo es conectarse a otros, es razonable establecer un conjunto de *pruebas de caja negra* para evaluar el nivel de cumplimiento de los requisitos.

2.2.1.1. Information Security in the System Development Life Cycle

El proyecto *Information Security in the System Development Life Cycle* (IS SDLC) del NIST estadounidense está enfocado a identificar, analizar y revisar el papel de la seguridad en el ciclo de vida de los sistemas de información, utilizando para ello modelos existentes en los que se identificarán los requisitos y características de seguridad necesarias.

Aunque el proyecto no ha producido ninguna documentación pública, si se puede acceder a los objetivos generales y a la descripción del enfoque que se utilizará en este proyecto. Dicho enfoque consiste en identificar las fases comunes a los diferentes ciclos de vida de los sistemas de información, para analizar cuáles de las tareas que se llevan a cabo en cada una de ellas tiene relación con la seguridad, de forma que sea posible identificar las necesidades y objetivos de seguridad que deben ser satisfechos. Aspectos clave de este proyecto son la identificación de los servicios de seguridad requeridos en cada fase, así como la evaluación de la necesidad de dichos servicios en relación con el papel del sistema de información dentro de la organización que hará uso de él. También es un aspecto importante la identificación de los diferentes aspectos regulatorios y normativos que afectan a la forma en que dichos servicios de seguridad se ofrecen, así como la identificación de las amenazas a las que deberá hacer frente el sistema. Todos estos aspectos serán analizados y el proceso de análisis y evaluación de estos servicios de seguridad en particular, y de los objetivos y necesidades a los que hacíamos referencia en general, será integrado en las diferentes fases del ciclo de vida del sistema de información.

Como podemos ver, este proyecto se encuentra en la línea de las metodologías de evaluación del software mencionadas en el apartado 2.2.1, en las que el proceso se centraba únicamente en la fase de desarrollo dentro del ciclo

de vida. Aunque este proyecto abarca todo el ciclo de vida, la visión general es la misma que la de las metodologías ITSEC, TCSEC y los Common Criteria, abstrayéndose de la implementación concreta de cada mecanismo de seguridad, y centrándose en el tratamiento que se le dan a esas necesidades dentro de la planificación del ciclo de vida del sistema de información. Un enfoque similar al de este proyecto es el que se lleva a cabo en la *Operationally Critical Threat, Asset, and Vulnerability Evaluation*^{SM} (OCTAVE$^{\text{SM}}$, [4]), desarrollada por el Instituto de Ingeniería del Software de la Carnegie Mellon University, en Estados Unidos.

2.2.1.2. Automated Security Functional Testing

Por su parte, el proyecto *Automated Security Functional Testing* (ASFT), también llevado a cabo en el NIST, tiene por objetivo el desarrollo formal de un modelo de especificaciones del comportamiento de las funciones de seguridad que sirva de base para automatizar el proceso de evaluación de dichas funciones, incluyendo la generación de vectores de prueba, generación de código de prueba y análisis de los resultados.

Entre los productos que han surgido de este proyecto se encuentra el marco de trabajo de automatización de pruebas (*Test Automation Framework*), que ayuda a automatizar el proceso de evaluación de un sistema al proporcionar herramientas que permiten diseñar modelos funcionales, analizar dichos modelos, generar código que evalúe dicho modelo y llevar a cabo las pruebas generadas, analizando posteriormente el resultado de dichas pruebas. Sobre este marco de trabajo, la herramienta TAF-SFT permite llevar a cabo el mismo proceso centrándose en las funciones de seguridad del sistema.

Como podemos observar, este proyecto lleva a cabo un doble análisis de la seguridad en el sistema evaluado:

- Por un lado, lleva a cabo un **análisis formal** sobre la funcionalidad que se pretende obtener, validando su correcto diseño y el comportamiento previsto para el sistema al utilizar dicha función.
- Por otro lado, estudia el **comportamiento** de la implementación, al evaluar con el código generado las funcionalidades que aparecen en el modelo del sistema.

Sin embargo, tal y como ocurre con otras propuestas ya mencionadas anteriormente, para llevar a cabo los análisis hacia los que está enfocado este proyecto es necesario conocer detalladamente el funcionamiento interno del sistema de información. Esto convierte los análisis en pruebas de caja blanca, lo que no permite evaluar de forma fiable la funcionalidad resultante de cara a una entidad externa al sistema.

2.2.2. Guías de configuración de la seguridad

A partir de la validación durante el proceso de desarrollo, surgió la necesidad de evaluar la seguridad de los sistemas de información durante el periodo operativo de dicho sistema, de forma que los análisis que se llevaron a cabo durante el proceso de desarrollo se complementen con el estudio de la seguridad del sistema una vez desplegado en el entorno en el que debe realizar su función.

Durante el proceso de despliegue del sistema de información los análisis toman la forma de guías de configuración segura de los diferentes sistemas, permitiendo evaluar si todos los pasos necesarios para que la seguridad del sistema no se vea comprometida por defectos en los procesos de instalación y configuración. Algunas de las listas de configuración más utilizadas en la actualidad tienen un carácter artesanal en la mayoría de las casos, como puede ser el caso de [27] y [28]. Sin embargo, en este apartado revisaremos algunas de las guías de configuración de la seguridad más representativas en la actualidad, que cuentan con un respaldo mayoritario por parte de la comunidad internacional y que cuentan con mayor grado de formalidad que las guías anteriormente citadas.

2.2.2.1. Guías de configuración de la seguridad

NSA

La Agencia Nacional para la Seguridad estadounidense (NSA) publica guías de configuración de la seguridad en múltiples dispositivos de red y sistemas operativos, con guías especializadas en tecnologías que, bien por su popularidad, bien por problemas detectados por su diseño, pueden ser origen de graves problemas de seguridad tanto para particulares como para compañías en las que se instalen dichos sistemas. Dichas guías son puestas a disposición del público, y pueden ser consultadas y descargadas desde [118], encontrándose en la actualidad guías para la configuración segura de sistemas operativos, aplicaciones, servidores de bases de datos, encaminadores, conmutadores, arquitecturas de Voz sobre IP, servidores y navegadores web, y redes inalámbricas 802.11.

Adicionalmente, dentro de este conjunto de guías de configuración también se encuentran documentos que discuten la situación de algunas tecnologías de seguridad en entornos concretos, o la forma de aplicar todo el conjunto de mecanismos y herramientas de seguridad de manera que maximice la seguridad de nuestro sistema de información. Ejemplos de estas guías son las recomendaciones para implantar la seguridad en profundidad, las recomendaciones a la hora de redactar informes de seguridad o de vulnerabilidades, y discusiones acerca del papel de los diferentes tipos de cortafuegos

en las redes corporativas.

Estas guías de configuración ofrecen una detallada recopilación de tareas y comprobaciones a llevar a cabo para validar la correcta configuración de la seguridad en el sistema, servicio o dispositivo que se evalúa, junto con los procedimientos para corregir posibles errores en dicha configuración. Estas recopilaciones se acompañan de una discusión sobre el problema de seguridad al que se está haciendo frente, junto con referencias a bibliografía y documentación adicional con la que profundizar en el tema que se discute.

Por lo tanto, el papel que juegan estas guías de configuración es el de servir de base para un proceso manual de verificación de la seguridad en un sistema de información una vez éste ha sido implementado y se encuentra, bien en funcionamiento, bien dispuesto a pasar a modo operativo en breve. Estas guías llevan a cabo un análisis de la seguridad del sistema centrándose en aspectos de la configuración que ayuden a proporcionar el máximo nivel de seguridad mientras se ofrece el servicio al que dicho sistema está destinado.

CIS

Por su parte, el Centro para la Seguridad de Internet (*Center for Internet Security*, CIS), ofrece también guías de configuración de seguridad, a las que acompaña con un sistema de calificación y evaluación, de forma que sea posible evaluar y comparar el nivel de seguridad de dos sistemas de información diferentes. Aunque en un número menor que las guías de la NSA, las guías publicadas por el CIS cuentan con un documento parejo en el que se muestran el proceso de evaluación llevado a cabo con varios productos, comerciales o no, en el que se diseña una arquitectura de pruebas, y se llevan a cabo evaluaciones sobre aspectos muy concretos de la configuración de los dispositivos. Como resultado final, se obtiene una valoración global de los diferentes dispositivos o versiones de los sistemas, con la que es posible el nivel de adecuación a unas necesidades determinadas.

Otro aspecto importante de estos conjuntos de pruebas es que desde su concepción inicial definen al menos dos niveles de seguridad y estudian el impacto de cada uno de los aspectos analizados en cada guía de configuración para cada uno de los niveles de seguridad definidos, lo que permite considerar la importancia de cada aspecto de la configuración que se revisa en la guía de forma relativa al nivel de seguridad objetivo para el sistema.

Adicionalmente los conjuntos de pruebas del CIS se complementan con herramientas de evaluación más o menos automatizadas (como por ejemplo, las configuraciones pre-generadas para utilizar con bastille-linux ([14]), como medio de configurar un sistema con un determinado nivel de seguridad de forma automatizada, o la especificación de las pruebas a llevar a cabo en lenguaje XCCDF (del que se hablará con mayor detenimiento en el apartado

2.2.3.1), lo que permite su utilización por parte de cualquier herramienta con soporte para dicho lenguaje.

Como podemos ver, en el caso del CIS las guías de configuración no se limitan únicamente a una recopilación de aspectos que deben ser tenidos en cuenta a la hora de configurar un sistema de información, sino que se incluyen herramientas y mecanismos para automatizar el análisis y configuración de esos aspectos, permitiendo así una mayor eficiencia a la hora de llevar a cabo estos procedimientos en un número elevado de sistemas.

2.2.3. Conjuntos de pruebas de seguridad

La evolución lógica de las guías de configuración nos lleva a la automatización del proceso de evaluación y pruebas de un sistema de información, en el que el sistema se evalúa automáticamente, de forma local o remota, según el tipo de análisis que se desee llevar a cabo, comprobando por un lado si la configuración utilizada en la actualidad puede ser utilizada por un potencial atacante para quebrantar la seguridad del sistema, y por otro lado para comprobar si ataques, vulnerabilidades y vectores de ataque que hayan aparecido desde la puesta en funcionamiento del sistema de información tienen efecto en el mismo. Este tipo de análisis de caja negra es muy utilizado en la actualidad para llevar a cabo pruebas de penetración en sistemas de información, ya que los resultados que ofrecen se refieren a los sistemas como cajas negras, tal y como reaccionan a mensajes y eventos de otros sistemas.

En este ámbito de la automatización de las pruebas de seguridad podemos encontrarnos en la actualidad tres líneas de trabajo diferentes pero complementarias: La primera de ella, apoyada sobre todo desde las organizaciones encargadas de la estandarización de métodos y procedimientos, lleva a cabo trabajos para diseñar medios y herramientas que permitan el intercambio de la información relativa a estas pruebas independientemente de la plataforma y sistema de pruebas que se utilice; por otro lado, el desarrollo de herramientas automatizadas de evaluación de la seguridad de los sistemas de información, especialmente de aquellos que ofrecen servicios en red, es otro área en la que se están llevando a cabo importantes avances en los últimos años; por último, la investigación en el área de la auto-evaluación de la seguridad por parte de los sistemas de información es otro aspecto en el que se están llevando a cabo importantes avances en los últimos años.

A continuación se revisarán las más importantes contribuciones de estos últimos años en cada una de las tres líneas de trabajo.

2.2.3.1. Extensible Configuration Checklist Description Format

El lenguaje extensible de definición de listas de configuración (*Extensible Configuration Checklist Description Format*, XCCDF, [157]) es un lenguaje

descriptivo diseñado para la definición de listas de seguridad como las descritas en el apartado 2.2.2.1, conjuntos de pruebas automatizados (como los que se estudiarán en el siguiente apartado) y otros documentos relacionados (tales como guías de usuario o manuales). El desarrollo de XCCDF depende del NIST estadounidense, aunque otras agencias gubernamentales (como la Agencia Nacional de Seguridad) se encuentran también involucradas en dicho proceso.

Este lenguaje se basa en tecnologías existentes y extendidas como XML para representar colecciones estructuradas de reglas de configuración para la evaluación de la seguridad en determinados sistemas de información. Este lenguaje permite y facilita el intercambio de información entre sistemas de evaluación automatizados, así como la interpretación de documentos generados y la modificación de las especificaciones para dar soporte a necesidades particulares. De esta forma se pretende proporcionar unas bases uniformes sobre las que definir pruebas y evaluaciones de seguridad particulares, así como conjuntos de pruebas más extensos y complejos, y por ello la actual especificación define un modelo para almacenar los resultados de pruebas de conformidad obtenidos a partir de herramientas automatizadas.

Los objetivos que se plantean para este lenguaje en particular, y para esta línea de investigación en general son, principalmente, los siguientes:

- Proporcionar los mecanismos para definir reglas y configuraciones de seguridad de forma eficiente y ágil.
- Permitir la interoperabilidad de diferentes herramientas, libres y comerciales, en cuanto a lo que la definición de reglas y configuraciones se refiere.
- Facilitar la validación de un conjunto de reglas y configuraciones contra una base de datos de políticas (que pueden venir dadas por requisitos formales, legales o de otra naturaleza).
- Habilitar la comparación de diferentes herramientas de evaluación de la seguridad, así como de los resultados obtenidos por dichas herramientas.

2.2.3.2. Herramientas automatizadas de evaluación de la seguridad

Como una segunda línea de investigación en el ámbito de los conjuntos de pruebas de seguridad se presentan las herramientas automatizadas para la evaluación de la seguridad. Estas herramientas llevan a cabo un conjunto de pruebas que evalúan aspectos concretos de la seguridad del sistema que evalúan, generando finalmente un informe en el que se recogen los resultados de las pruebas llevadas a cabo y otra información útil (como por ejemplo,

referencias a los problemas de seguridad que se hayan podido detectar, como informes de vulnerabilidades o la base de conocimiento del fabricante del sistema).

Aunque estas herramientas (sobre todo las que llevan a cabo evaluaciones de la seguridad de los servicios de comunicaciones) se han utilizado tradicionalmente en el ámbito de los atacantes para obtener información acerca de posibles vectores de ataque a un sistema de información, su uso no está restringido a las actividades de ataque, sino que su utilización como herramienta de detección de vulnerabilidades está muy extendida tanto entre responsables de los sistemas como entre los profesionales del sector.

Entre las aplicaciones más representativas en la actualidad de estas herramientas de evaluación de la seguridad podemos encontrar *Nessus* y *Sara*, ambos derivados de la *Security Administrator's Tool for Analyzing Networks* (SATAN), desarrollado en 1.995. Mientras que Sara ([147]) se centra en la realización de pruebas que no dañen el sistema evaluado (por lo que no incluye ninguna prueba de denegación de servicio) con el fin de que las pruebas puedan realizarse sobre sistemas operativos sin el riesgo de interrumpir el servicio, Nessus ([139]) por su parte centra su potencial en su mayor base de datos de ataques y vulnerabilidades, por lo que es capaz de llevar a cabo un análisis más completo de los sistemas evaluados.

La otra gran diferencia fundamental entre ambas herramientas es la forma en la que la información sobre las vulnerabilidades es almacenada e incorporada a la aplicación: mientras que Sara utiliza la base de datos de vulnerabilidades mantenida por el NIST ([117]), siendo compatible con el formato CVE de los informes de ficha base de datos, Nessus utiliza un lenguaje propio denominado NASL (*Nessus Attack Scripting Language*) en el que se codifican todos los parámetros de cada vulnerabilidad que debe ser evaluada.

El modo de funcionamiento de ambas herramientas es muy similar, utilizando una arquitectura de red como la que se puede ver en la Figura 2.6, en la que un sistema en el que se ejecuta la herramienta de evaluación de la seguridad lleva a cabo pruebas de seguridad, remotas si se llevan a cabo en otros sistemas, y locales si se llevan a cabo en el propio sistema. Las pruebas se llevan a cabo siguiendo las políticas de temporización y paralelización que se hayan definido, y al finalizar el proceso de evaluación se obtiene un informe de resultados.

En la actualidad, uno de los mayores problemas que tienen estas herramientas es la gran cantidad de falsos positivos que generan, debido principalmente a la falta de rigor formal en la realización de las pruebas, lo que lleva a la interpretación de toda aquella respuesta imprevista como un error del sistema, lo que se asocia a la vulnerabilidad que se lleva a cabo en el momento en que se recibe la respuesta.

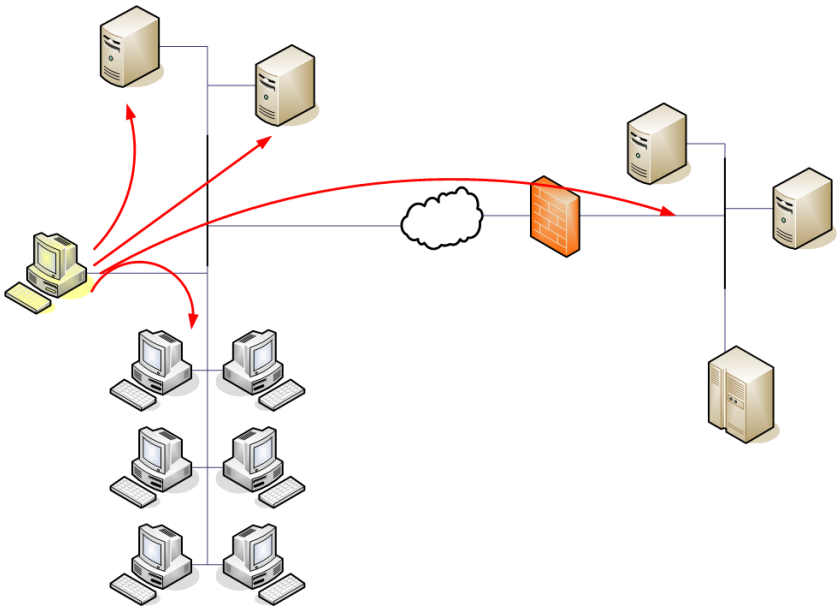

Figura 2.6: Modelo de utilización de herramientas automatizadas. Su instalación en un equipo permite analizar otros equipos de la misma red (análisis interno), o incluso de otras redes (análisis externo)

2.2.3.3. Automated Security Self-Evaluation Tool

La última línea de investigación que se analizará en el área de los conjuntos de pruebas de seguridad es la de la auto-evaluación por parte de los sistemas de información acerca de sus propias vulnerabilidades y problemas de seguridad, en lugar de depender de sistemas externos que lo evalúen. El proyecto más representativo en éste ámbito es la Herramienta de Auto-Evaluación de Seguridad Automatizada (*Automated Security Self-Evaluation Tool*, ASSET, [115]), desarrollado por el NIST como consecuencia de la publicación del informe especial de seguridad 800-26 ([143]), en el que se proporcionan las líneas generales que deben regir los procesos de auto-análisis por parte de los sistemas de información.

En la actualidad el proyecto no se encuentra finalizado, por lo que únicamente es posible llevar a cabo la evaluación de un sistema de información a partir de cuestionarios realizados a los usuarios del sistema, que, junto con las reglas y políticas que definen los diferentes niveles y requisitos de seguridad, sirven para obtener los resultados de la evaluación de la seguridad del propio sistema de información.

2.2.4. Validación de protocolos de seguridad

Por lo que respecta a la evaluación de los protocolos de seguridad, en la actualidad podemos encontrar en la literatura científica dos enfoques diferentes, cada uno de ellos centrado en un aspecto diferente del protocolo de seguridad. El primero de estos enfoques se centra en el análisis del protocolo en sí mismo, estudiando el flujo de información entre las entidades, el intercambio de mensajes, las propiedades criptográficas de las herramientas utilizadas, etc. . . ., todo ello desde un punto de vista formal y teórico, sin analizar las implementaciones de ese protocolo. El segundo enfoque se centra en el análisis de las implementaciones de los protocolos de seguridad, y es un enfoque eminentemente práctico, que no intenta averiguar si el protocolo utilizado es seguro o no, sino que analiza cómo se ha implementado esa definición teórica del protocolo.

Cada uno de estos enfoques ha sido analizado y estudiado por científicos de todo el mundo, realizándose grandes avances tanto en uno como en otro sentido. Por este motivo se procederá a analizar el estado de la cuestión de cada uno de estos enfoques utilizando como guía un proyecto representativo de cada uno de ellos: el proyecto *AVISPA* para la validación formal de protocolos y el proyecto *IPsec-WIT* como "representante" de la validación de las implementaciones.

2.2.4.1. Validación formal del protocolo: Proyecto AVISPA

En lo tocante a la validación formal del protocolo, abundan hoy en día trabajos que hacen uso de estas herramientas (ya sean lógicas formales ([24]), modelos simbólicos ([22]), modelos de razonamiento ([74]), conjuntos de evidencias formales de la seguridad ([109]) o nuevos sistemas de verificación ([128])). Sin embargo, todas estas técnicas están orientadas a la validación del protocolo en sí mismo, mientras que lo que en esta tesis se desarrollará será un sistema para evaluar **la implementación del protocolo de seguridad**. Un ejemplo de este tipo de validaciones lo podemos encontrar en el proyecto AVISPA.

El proyecto AVISPA (*Automated Validation of Internet Security Protocols and Applications*) es un proyecto englobado en el V Programa Marco de la Unión Europea, dentro del programa de Tecnologías Emergentes y Futuras. Su ámbito de trabajo es la validación de los protocolos y aplicaciones de seguridad utilizados en Internet mediante un enfoque formal, de forma que los vectores de ataque y las posibles vulnerabilidades puedan ser detectados antes de que ese protocolo defectuoso haya sido implementado y puesto a funcionar en un entorno de producción. Para conseguir estos objetivos el proyecto AVISPA desarrolló nuevos lenguajes de alto nivel con los que es posible expresar las especificaciones de los protocolos de seguridad

(favoreciendo la transición desde el lenguaje formal utilizado en el estándar a otro lenguaje que pueda ser procesado automáticamente) que facilitaban el posterior análisis y evaluación.

Este proyecto se centraba en el análisis de los protocolos y las aplicaciones en sí mismos (es decir, del diseño del protocolo o de la aplicación), proporcionando como resultado información acerca de la bondad del diseño del protocolo o de la aplicación. El análisis llevado a cabo se basa en un conjunto de pruebas que tienen su origen en ataques existentes o vulnerabilidades conocidas de otros protocolos y aplicaciones de seguridad, a los cuales es sometido el protocolo o aplicación que se desea estudiar. En la medida en que estos tests requieren del acceso a los mecanismos internos de la aplicación o del protocolo para funcionar, es sensato calificar los mismos como pruebas de caja blanca.

El interfaz de usuario del proyecto AVISPA está basado en tecnologías web, tanto para el diseño de nuevos conjuntos de pruebas como para la evaluación de protocolos y aplicaciones. Las posibilidades que ofrece el interfaz van desde la edición de protocolos y las pruebas, hasta la revisión de los resultados de pruebas llevadas a cabo anteriormente. Estos resultados constan, por un lado, de los resultados de las pruebas en sí mismos, y por otro, de los recursos que han sido necesarios para llevar a cabo las pruebas.

Aunque el enfoque utilizado en el proyecto AVISPA es importante para la validación de los protocolos de seguridad, sus resultados sólo afectan al diseño del protocolo, sin que la implementación que se hace de ese diseño sea analizada en ningún momento. Esto quiere decir, entre otras cosas, que este tipo de análisis no puede proporcionar información acerca de la interconectividad entre las diferentes implementaciones.

2.2.4.2. Validación de la implementación: Proyecto IPsec-WIT

El otro enfoque principal que encontramos hoy en día en lo tocante a la validación de protocolos de seguridad se centra en el análisis de las implementaciones de los protocolos en lugar de analizar los protocolos, de forma que la información que se obtiene hace referencia a cómo se comporta la implementación frente a otros dispositivos y sistemas. Como podemos ver, el análisis que se lleva a cabo es totalmente diferente que en el caso anterior, ya que la información que se busca obtener se encuentra en un plano diferente al anterior. Como proyecto que ejemplifica las características de este enfoque se ha seleccionado el proyecto IPsec-WIT, del National Institute of Standards and Technology estadounidense.

IPsec-WIT ([120]) es un proyecto llevado a cabo por el NIST estadounidense a partir de una petición del IETF solicitando sistemas que llevasen a cabo pruebas de interoperabilidad para IPsec. Cuando el IETF constató los problemas que empezaban a surgir concernientes a la interoperabilidad de

las implementaciones de protocolos de seguridad lanzó esta solicitud para que se diseñasen e implementasen conjuntos de pruebas de interoperabilidad. IPsec-WIT (que quiere decir *IPsec Web Interoperability Test*) fue una de las propuestas que surgieron a partir de esa llamada, y tenía como objetivo el ser una herramienta de referencia con un interfaz web que probase la conformidad de una implementación con los estándares de IPsec.

IPsec-WIT se basaba en la implementación de referencia del NIST de los protocolos que componen IPsec (llamadas `PlutoPLus` y `Cerberus`), y ofrecía la posibilidad de probar la conformidad de las diferentes suites criptográficas para cada protocolo de IPsec, así como la ejecución del protocolo y la negociación de los diferentes parámetros en las diferentes fases del establecimiento de las asociaciones de seguridad. Todas estas pruebas se configuraban y arrancaban a través de un interfaz web que el NIST ofrecía en su servidor.

Este proyecto tuvo que hacer frente a varios problemas que hacen inviable su utilización en la actualidad. En primer lugar, las pruebas que se llevaban a cabo requerían de un gran nivel de reconfiguración (y por lo tanto, de intervención por parte del usuario/administrador) en el proceso, convirtiéndose ésta en una tara muy tediosa para las personas que debían llevarla a cabo (personas que en la mayoría de los casos debían decidir si las pruebas se llevaban a cabo o no). Adicionalmente, el proyecto ha sido interrumpido y no se trabaja más en él, y debido a que la plataforma utilizada no se ha hecho pública (tests, implementaciones de los protocolos de IPsec, etc....) no es posible tratar de dar continuidad al proyecto. Por último, los resultados obtenidos por la plataforma informaban acerca de la implementación que había sido probada de forma aislada, sin proporcionar información acerca de las posibilidades de interconexión con otros dispositivos o soluciones que podamos encontrar.

IPsec-WIT fue el primer[2] proyecto importante en abordar el problema de la interconectividad entre las diferentes implementaciones de los protocolos de seguridad, centrándose en las implementaciones de IPsec. En su contra, es importante destacar que carecía de algunas características de análisis y facilidad de uso que son necesarias actualmente, especialmente cuando las tareas a llevar a cabo requieren de un prolongado espacio de tiempo. Sin embargo, su gran contribución (además de ser el primer proyecto en abordar el problema) probablemente sea el proporcionar un interfaz web desde el que gestionar el conjunto de pruebas, ya que de esta forma se está utilizando un soporte que la gran mayoría de los dispositivos actuales son capaces de gestionar.

[2]De hecho, la única referencia a otro trabajo similar en cualquiera de los cuerpos de estandarización, en el que el objeto de evaluación sea la implementación del protocolo de seguridad en sí misma, y no el protocolo que se implementa, es un borrador de RFC ya caducado, en el que se establecían los criterios para la evaluación de implementaciones de Redes Privadas Virtuales ([155])

Como podemos ver, este enfoque complementa al anterior, en tanto que uno se preocupa de la seguridad del protocolo, llevando a cabo estudios teóricos sobre el mismo, mientras que el otro concentra sus esfuerzos en analizar cómo se ha trasladado esa definición teórica de un intercambio de información con ciertas propiedades que confieren seguridad a dicho intercambio a un dispositivo, sistema o software determinado. Ambos enfoques son necesarios para garantizar en la medida de lo posible la seguridad del sistema,

2.2.5. Metodologías de evaluación de la seguridad de sistemas de información

Una vez llevada a cabo la revisión del estado de la cuestión en aspectos concretos de la validación y evaluación de la seguridad, cabe preguntarnos por la existencia de metodologías que lleven a cabo una validación de los mecanismos de seguridad integrados en los sistemas de información. Actualmente podemos encontrar múltiples metodologías que llevan a cabo una evaluación de la seguridad en un sistema de información; sin embargo, nos encontramos con que la mayoría de estas propuestas resultan ser métodos (es decir, conjuntos de pasos a seguir) y no metodologías, ya que sus propuestas se centran en la especificación de pasos concretos a seguir para obtener información concreta con la que elaborar un informe final (como es el caso de la *Open Source Security Testing Methodology*, desarrollada por el ISECOM (*Institute for Security and Open Methodologies*), y que puede encontrarse en [68].

La única propuesta actualmente vigente en la que realmente se propone una metodología para llevar a cabo una evaluación de la seguridad de los sistemas de información es la Metodología de Revisión de la Seguridad (*Security Review Methodology*) desarrollada por la Agencia de Sistemas de Información para la Defensa (*Defense Information Systems Agency*, CISA), del gobierno estadounidense ([127]), y que pasaremos a estudiar con mayor detalle a continuación.

2.2.5.1. Metodología de Revisión de la Seguridad

Esta metodología describe el proceso de evaluación de sistemas de información (en concreto, aplicaciones software) en cuanto a su seguridad, con el fin de determinar que el sistema analizado:

1. No introducirá problemas de seguridad en la red en la que se instale

2. No requiere que se relajen las medidas de seguridad existentes para que pueda operar

Para poder llevar a cabo esta evaluación en la metodología se describen siete fases (inicio, documentación, configuración básica preliminar, instalación y configuración, pruebas de seguridad, pruebas funcionales, resultados) en las que dividir el proceso de evaluación, aunque la propia metodología admite que es posible alterar el orden o incluso omitir algunas de las fases si no se está interesado en la información que se obtiene de dicha fase. Asimismo, se definen dos niveles de evaluación diferentes: aquel en el que la evaluación se lleva a cabo sobre las especificaciones del producto, y aquella en la que la evaluación se centra en la identificación de problemas de seguridad que surjan del uso del sistema evaluado.

Para cada una de las diferentes fases en las que se divide el proceso, la metodología proporciona los objetivos a alcanzar en dicha fase, junto con las diferentes pautas de desarrollo de la fase de acuerdo al nivel de evaluación que se lleve a cabo. Adicionalmente, para cada elemento de decisión, procedimiento, o mecanismo que deba llevarse a cabo o utilizarse se describen situaciones comunes y recomendaciones acerca de cómo deben ser resueltas. Estas situaciones van desde aspectos técnicos ("en qué sistema operativo se instalará el producto a probar") hasta legales ("el responsable de la evaluación firmará los acuerdos de confidencialidad establecidos en las normas reguladoras").

El proceso completo de evaluación, tal y como se describe en la metodología, consta de los siguientes pasos:

1. Inicio
   - *a*) Determinación del tipo y alcance de la evaluación
   - *b*) Obtención de los recursos humanos y técnicos necesarios
   - *c*) Firma de documentos legales y acuerdos de confidencialidad
   - *d*) Creación del soporte de datos necesario para la evaluación
2. Documentación
   - *a*) Determinación del nivel de seguridad del producto según Common Criteria
   - *b*) Lectura de la documentación del producto
   - *c*) Búsqueda de vulnerabilidades del producto en Internet
   - *d*) Instalación en un sistema de pruebas para familiarizarse con el producto
   - *e*) Documentación de los resultados preliminares
3. Configuración Básica Preliminar

   a) Desplegar una plataforma conforme a STIG[3]
   b) Análisis de la seguridad del sistema tras la instalación de la plataforma

4. Instalación y Configuración

5. Pruebas de Seguridad

   a) Realización de las pruebas de seguridad de la aplicación
   b) Comparación de las configuraciones iniciales y finales
   c) Análisis de la seguridad del sistema tras la instalación y configuración de la aplicación

6. Pruebas Funcionales

   a) Desarrollo de las pruebas funcionales
   b) Ejecución de las pruebas funcionales

7. Resultados

   a) Generación de los informes de resultados
   b) Revisión de la calidad de la evaluación
   c) Generación y diseminación del informe final de la evaluación
   d) Liberación del equipamiento y personal

Como podemos observar, el objetivo de la metodología no es describir los pasos individuales necesarios para llevar a cabo una evaluación de un producto determinado, sino que desgrana el proceso de evaluación genérico para cualquier sistema que deba evaluarse, dejando al evaluador la libertad suficiente para que pueda adaptarse convenientemente al producto evaluado, e incluyendo los necesarios controles de calidad para asegurarse de que la evaluación se lleva a cabo de acuerdo a las pautas establecidas por políticas internas a la organización.

2.3. Evaluación del rendimiento

Al estudiar las soluciones para evaluar el rendimiento de un sistema de comunicaciones seguro, podemos encontrar varias metodologías de evaluación del rendimiento de sistemas de comunicaciones, tales como [53] (donde se proponen métodos para desarrollar entornos de pruebas adaptados a

[3]Una plataforma conforme a STIG es aquella en la que todos los sistemas presentes en dicha plataforma han sido instalados conforme a las guías técnicas de configuración de la seguridad (STIGs) de la Agencia de Sistemas de Información para la Defensa, el mismo organismo que desarrolla esta metodología

nuestras necesidades particulares), [51] (en la que se utilizan pares cliente-servidor para llevar a cabo las mediciones) o [23] (donde se exponen métodos para utilizar entornos de pruebas personalizados y comparar los resultados con otros bancos de pruebas). Sin embargo, el denominador común de estas metodologías es que no están pensadas para ser utilizadas en sistemas de comunicaciones que incluyan la seguridad en sus características. Como hemos visto anteriormente, la alta variabilidad que presentan los resultados de rendimiento al llevar a cabo operaciones criptográficas hacen que la extrapolación de datos no sea recomendable, lo que convierte algunas de las propuestas en inviables.

El otro gran problema que se presenta con las metodologías tradicionales de evaluación del rendimiento es que, dado que el mero establecimiento de una conexión sin seguridad no supone una sobrecarga importante para el servidor, el impacto que el volumen de conexiones simultáneas tiene en el rendimiento del sistema no se evalúa, pese a la importancia de este factor en las comunicaciones seguras. Alrededor del establecimiento de la conexión existen una conjunto de parámetros que es importante conocer para obtener información relevante acerca del rendimiento.

El problema de que estas metodologías están pensadas para sistemas de comunicaciones que no incluyen la seguridad en sus características hace que en las ocasiones en las que se intentan utilizar estas metodologías en sistemas con seguridad, los resultados carezcan del rigor científico necesario. Por ejemplo, al intentar medir la sobrecarga que introduce la securización del canal de comunicaciones con respecto al canal de comunicaciones no seguro, es común realizar un número mínimo de medidas y de ahí extrapolar el resto de datos. Sin embargo, dado el amplio espectro de tecnologías y anchos diferentes disponibles hoy en día, ésta no es una opción válida para un análisis exhaustivo de la implementación del protocolo seguro.

Analizando las propuestas de la comunidad científica en los últimos años, podemos encontrar que las bases de la medición actual del rendimiento para protocolos y dispositivos de red se remonta a 1.996, año en el que se publicaron las propuestas para el análisis del rendimiento en aplicaciones cliente / servidor ([52]) y las herramientas `netpipe` ([141]) y `NetPerf` ([81])se consideran estables. En estos artículos y herramientas se determinaban que los aspectos relevantes de la medición de rendimiento eran, básicamente, el ancho de banda que se podía ofrecer, la sobrecarga que se produce en los diferentes protocolos al variar ciertos parámetros de la transmisión, la latencia de la red y la pérdida de tramas producida en casos de saturación. Para que los resultados que se obtienen sean fiables y averiguar cuál es la variación del rendimiento entre las diferentes configuraciones, se establecen modificaciones del tamaño del paquete de datos a enviar, se evalúa el comportamiento con transferencias de flujos y con transferencias de bloques de datos, etc. . . .

A partir de ese momento se pueden observar dos tendencias claramente diferenciadas: por un lado se han ido desarrollando nuevas herramientas para obtener la información de rendimiento, que optimizan o mejoran las técnicas propuestas en 1.996, como es el caso de la herramienta `iPerf` ([73]), o propuestas como [156]. Estas propuestas mejoran las características, el tiempo de ejecución o los recursos necesarios para llevar a cabo el análisis de rendimiento propuesto por otros autores anteriormente. Por otro lado, nuevas propuestas en cuanto a la medición del rendimiento se han ido sucediendo, aunque el interés se ha trasladado de la medición en si misma a la medición en determinados entornos o protocolos, y a la medición del rendimiento de los procesadores de red.

Por estos motivos, podríamos decir que el último gran trabajo en cuanto al análisis de los parámetros que es necesario evaluar para obtener información fiable y realista acerca del rendimiento de red ha sido la publicación por parte del IETF de la RFC *Benchmarking Methodology for Network Interconnect Devices* ([19]), en la que no sólo se analizan cuáles son los parámetros que es necesario evaluar, sino también qué problemas suelen darse al tomar las medidas (por ejemplo, realizar una medición en un "pico" de tráfico, lo que originaría información falsa) y proporciona métodos para evitar estos problemas. Sin embargo, en los últimos años el propio IETF está revisando y actualizando las metodologías de evaluación del rendimiento, aunque en lugar de proporcionar una gran metodología para todo tipo de dispositivos y tráfico, está optando por publicar múltiples metodologías que se adapten a las particularidades de un protocolo, grupo de protocolos o tipo de dispositivos ([100], [15], [69]) (en la línea de la segunda tendencia mencionada anteriormente.

Al estudiar las propuestas de evaluación del rendimiento que han surgido de esa segunda tendencia podemos ver cómo las propuestas que evalúan el rendimiento de los procesadores de red (como por ejemplo `NpBench` ([96]), `CommBench` ([151]) o `NetBench` [102]) se centran en el rendimiento del procesador, por lo que su enfoque es de muy bajo nivel, evaluando el tipo y cantidad de instrucciones de bajo nivel que ejecuta un procesador de red en determinadas situaciones. Sin embargo, dada la proximidad de este área con la evaluación del rendimiento en procesadores "tradicionales", podemos ver cómo su evolución ha sido más rápida, disponiendo de algunas metodologías de evaluación del rendimiento que sirven de guía y referencia para llevar a cabo este tipo de mediciones (por ejemplo, [148]).

La otra "rama" de la evaluación del rendimiento que ha surgido de esa segunda tendencia que mencionábamos anteriormente se ha centrado en mejorar la medición del rendimiento en determinadas redes, arquitecturas o protocolos, estudiando cuáles son las particularidades de la red en la que se va a llevar a cabo el estudio, y así mejorar tanto la captura de datos como la interpretación de resultados. Algunos resultados de esta línea de inves-

tigación podemos verlos en [61], donde se analizan la influencia de utilizar comunicaciones TCP en paralelo en redes con poca fiabilidad, [97], donde se proponen técnicas para medir el rendimiento de la red en entornos Grid, o [20], donde se estudia cuál es la sobrecarga que impone en las comunicaciones el protocolo CONFIDANT. En esta misma línea podemos citar también los propios trabajos del IETF en los últimos años ([100], [15]), y algunos estudios realizados referentes a la sobrecarga que introducen los protocolos de seguridad ([7] y [8] analizan SSL/TLS, mientras que [104] proporciona una comparativa entre IPsec y SSL/TLS).

Una última tendencia que está cobrando fuerza en los últimos años es la de integrar la medida del rendimiento de una red o protocolo con estudios acerca del uso que se hace de dicha red o protocolo, con el fin de facilitar las previsiones de crecimiento y evolución futura de las redes. Algunos ejemplos de este tipo de trabajos los podemos encontrar en [12], donde se utiliza la combinación de perfiles de usuarios y rendimiento de la red para planificar la aplicación de parámetros de calidad del servicio, [152], donde dinámicamente se construyen modelos basados en la utilización y rendimiento de la red, de forma que sea posible predecir el comportamiento tanto de la red como de los usuarios en un futuro, y [13], donde se utilizan los perfiles de uso de dispositivos móviles para limitar las funciones disponibles y así eliminar posibles puntos de ataque.

2.3.1. Evaluación del rendimiento de los protocolos de seguridad

Centrándonos en la evaluación del rendimiento de los protocolos de seguridad, tras investigar en la literatura científica nos encontramos con que todos los trabajos son de reciente factura, encontrando los estudios más antiguos en 1.999, cuando Apostolopoulos et al. realizan el primer análisis a fondo del protocolo SSL ([8]), estudio que repitieron en 2.000 sobre TLS ([7]). Estos estudios coinciden con el momento en que la cantidad de dispositivos, protocolos, redes, aplicaciones, etc.... se incrementa exponencialmente, haciendo que estudios anteriores sobre el rendimiento de la red y los dispositivos de red pierdan parte de su validez, al no tener en cuenta las características de los nuevos dispositivos, protocolos y aplicaciones.

A partir de este momento asistimos a una preocupación generalizada por estudiar cuál es el rendimiento de los protocolos y arquitecturas de seguridad, tanto en redes genéricas (centrándose en esos casos en la influencia que tienen los diferentes aspectos del protocolo de seguridad (como por ejemplo, la autenticación) en el rendimiento de todo el protocolo (como el análisis realizado sobre el impacto del uso de clave pública en Kerberos en [63])) como en las nuevas redes que se expandían en esos años (por ejemplo, las redes de teléfonos móviles GSM, GPRS y similares en [64]).

Sin embargo, todos estos estudios se caracterizan por aplicar a protocolos o arquitecturas de seguridad las metodologías y conjuntos de pruebas de rendimiento que se aplican a protocolos y arquitecturas no seguros, lo que hace que los resultados nos proporcionen información asimilable para topologías de red, dispositivos, equipos o algoritmos diferentes de los que se utilizaron para llevar a cabo el estudio.

Adicionalmente, en los últimos años hemos podido observar cómo la problemática que representan los protocolos de seguridad al evaluar el rendimiento ha sido el centro de atención de múltiples instituciones, compañías e investigadores. De esta forma hemos podido ver cómo el IETF comenzaba a desarrollar algunas metodologías de evaluación de rendimiento para dispositivos de red con servicios de seguridad ([69]), cómo los protocolos de seguridad eran analizados para comprender cuáles son las particularidades de cada operación y mensaje que se transmite, etc....

Como resultado de esta nueva mentalidad, la evaluación del rendimiento de los protocolos de seguridad se está realizando en la actualidad de forma más cauta y elaborada. En el caso de IPsec, los estudios más elaborados acerca del rendimiento son "*IPSec Virtual Private Networks: Conformance and Performance Testing*" de 2.003 ([82]), y "*Tutorial on NPF's IPsec Forwarding Benchmark*" ([26]) y "*Validating IPsec Network Security Devices*" ([146]), de 2.004 ambos.

El primer estudio plantea la necesidad de llevar a cabo análisis de conformidad con el estándar y de rendimiento a las implementaciones IPsec. Aunque en el documento se plantea la necesidad de llevar a cabo ambos análisis, el enfoque se decanta sobre todo por los análisis del rendimiento de la implementación, limitando el análisis de la conformidad a las respuestas que la implementación puede dar como receptor de la solicitud de establecimiento de nuevos túneles IPsec, al tiempo que limita las pruebas que se llevan a cabo al análisis de mensajes concretos, sin llegar nunca a evaluar el completo desarrollo de los protocolos. Adicionalmente, en el estudio del rendimiento que se propone en ningún momento se tiene en cuenta la influencia de factores como el ancho de banda dedicado a cada túnel criptográfico o la influencia de la arquitectura de red utilizada para el desarrollo de las pruebas.

En el segundo de esos estudios podemos ver cómo ya se incluye la necesidad de tener en cuenta los parámetros del tráfico que se va a proteger con IPsec, ya que las operaciones de cifrado y descifrado no son equivalentes en sus requerimientos. También se incluye en este trabajo la necesidad de evaluar la latencia del protocolo, ya que el establecimiento de la conexión ahora implica la negociación de una clave mediante técnicas criptográficas, lo que puede significar un retardo importante.

El último de los estudios es más completo incluso que los dos anterio-

res, ya que en él se tienen en cuenta otros aspectos de IPsec, tales como la capacidad de establecer nuevos túneles criptográficos para proteger la información (aspecto que no se trata en el primer estudio), la cantidad máxima de túneles que el dispositivo o equipo puede soportar simultáneamente, o el impacto de utilizar *Perfect Forward Secrecy*.

Sin embargo, este tercer estudio también adolece de un análisis en profundidad de los escenarios en los que habitualmente se despliega una arquitectura IPsec, y no tiene en cuenta aspectos importantes para el rendimiento. Por ejemplo, uno de los mayores problemas que se pueden encontrar radica en que la medición de la cantidad de túneles criptográficos que se pueden establecer no tiene en cuenta cuánto tráfico se está transmitiendo por los túneles ya establecidos, o el impacto de realizar las conexiones en ráfagas (típicas en las redes corporativas a las horas de inicio de los turnos de trabajo, por ejemplo).

Por todos estos aspectos nos encontramos en posición de decir que, aunque en los últimos años se ha avanzado mucho en la evaluación del rendimiento de los dispositivos cuando se utilizan protocolos de seguridad, aún queda trabajo por hacer en lo referente al análisis de las características que afectan al rendimiento y que, por lo tanto, deben ser analizadas con mayor rigurosidad.

2.4. Resumen

En este capítulo hemos llevado a cabo un breve análisis de los estándares existentes en el ámbito de la seguridad, en el que hemos profundizado en los protocolos y arquitecturas de seguridad más extendidos en la actualidad. Igualmente se ha analizado el estado actual de los estándares de herramientas criptográficas y otras metodologías, arquitecturas, marcos de trabajo y propuestas que han sido estandarizados por alguno de los organismos de estandarización existentes, tanto a nivel internacional como de forma local para un país o territorio concreto.

Posteriormente hemos realizado un análisis del estado de la cuestión en lo referente a la validación de la seguridad, partiendo de la validación de los mecanismos de seguridad durante el proceso de desarrollo, para continuar analizando las guías de configuración de la seguridad en sistemas de información (correspondientes a la fase de despliegue en el ciclo de vida), y a las herramientas automatizadas de evaluación de la seguridad, correspondientes a la fase de mantenimiento del sistema de información. Posteriormente nos hemos centrado en las aportaciones de la comunidad científica en cuanto a la validación de protocolos de seguridad, para concluir esa sección estudiando las propuestas existentes de metodologías de evaluación de la seguridad en sistemas de información.

Por último, hemos revisado las principales contribuciones en el ámbito de la medición del rendimiento de protocolos y dispositivos de red, analizando las características de las principales propuestas y estudiando por qué los modelos tradicionales de medición del rendimiento no son adecuadas para evaluar el rendimiento de implementaciones de protocolos de seguridad. Finalmente, hemos dirigido nuestra atención a las propuestas más novedosas en lo referente a la medición del rendimiento de protocolos de seguridad, revisando sus aportaciones y señalando las carencias de cada una de ellas.

Capítulo 3

Análisis de la Conformidad con el Estándar

3.1. Introducción

En este capítulo se procederá a exponer cuáles son las características que es necesario evaluar para realizar un completo estudio acerca de la conformidad con el estándar de implementaciones de protocolos y arquitecturas de seguridad. Una vez identificadas estas características, se procederá a describir de qué forma se debe lleva a cabo su evaluación, estudiando los diferentes métodos, herramientas y técnicas que es posible utilizar para llevar a cabo los análisis necesarios, presentando las ventajas y los inconvenientes de cada uno de ellos y otros problemas que pueden surgir al llevar a cabo dicha evaluación de la conformidad con el estándar.

3.2. Identificación de aspectos de conformidad en protocolos y arquitecturas de seguridad

Como se ha comentado anteriormente, el origen de un elevado número de los problemas que se presentan en las implementaciones de los protocolos de seguridad se encuentra en las "mejoras" introducidas en dichas implementaciones (por ejemplo, el uso exclusivamente de suites criptográficas ligeras en lugar de las definidas en los estándares para cada protocolo). Por lo tanto, es necesario un estudio de los documentos en los que los protocolos, mecanismos, herramientas, etc... utilizados en los protocolos y arquitecturas de seguridad se definen, para así proceder a identificar a los factores que resultan vitales al evaluar la conformidad de una implementación con el estándar.

Tras llevar a cabo un análisis de las especificaciones de protocolos y

mecanismos de seguridad se concluye que aquellos aspectos que deberán ser evaluados a la hora de estudiar la conformidad con los estándares cuentas con las siguientes características:

- *Rigurosa definición de las estructuras de datos* a utilizar
- *Gran dependencia de la exactitud de la implementación para un correcto funcionamiento*
- *Uso intensivo de determinados mecanismos durante el desarrollo de la arquitectura de seguridad*, sin posibilidad de utilizar alternativas, y
- *Obligación de que ambas partes utilicen los mismos parámetros.*

Por *manejo de estructuras de datos* entendemos la utilización de estructuras de datos en las que no se permite margen de libertad al preparar los datos de entrada o en la representación de los datos de salida. Por ejemplo, en los cifradores de bloque los datos de entrada deben ser bloques de datos de un tamaño determinado, con una clave de cifrado de un tamaño concreto, y la salida será un bloque de datos de un tamaño especificado. Si la implementación del cifrador no prepara correctamente los datos de entrada o no reserva memoria suficiente para almacenar el bloque de datos de salida se producirá un error al utilizar ese cifrador que ha sido implementado incorrectamente. Del mismo modo, las estructuras de datos que conforman los diferentes campos de datos con los que operan los protocolos de seguridad han de seguir unas reglas estrictas acerca del modo en que se opera con esos campos: formatos, codificación, valores de relleno, etc.... Cualquier implementación que modifique la forma de utilizar esos campos de datos con respecto a lo especificado en el estándar del protocolo correspondiente generará problemas de interoperabilidad con otras implementaciones que traten los campos tal y como se especifica en el estándar.

La *dependencia de la exactitud de la implementación* hace referencia a la necesidad de las herramientas (especialmente las que poseen un alto componente formal, de que las implementaciones de dichas herramientas sean perfectamente conformes a la especificación, ya que en caso contrario pueden perderse las propiedades que convierten dicha herramienta en idónea para una labor determinada. Por ejemplo, un protocolo de seguridad normalmente es analizado teóricamente para prevenir filtrados de información a terceras partes; sin embargo, una implementación deficiente puede hacer posible que un atacante recabe información que debería estar protegida. Del mismo modo, el desarrollo de un protocolo de seguridad consta de determinados pasos encaminados al intercambio ordenado y seguro de información, y dichos pasos están definidos estrictamente en la especificación del protocolo. Una implementación que no siga esos pasos especificados se encontrará con problemas a la hora de establecer canales seguros con otras

implementaciones que sigan el desarrollo del protocolo como se define en el estándar.

El *uso intensivo de un mecanismo durante el desarrollo de la arquitectura de seguridad* implica que el factor (ya sea una herramienta, una propiedad que debe preservarse o un conjunto de datos concretos) es utilizado bien en múltiples fases del protocolo o arquitectura, bien en una fase pero de forma intensiva, por lo que el impacto de dicho factor en el desarrollo de cada una de las fases y de la arquitectura en general es muy importante. El ejemplo más claro es la implementación de los cifradores, ya que estas herramientas se utilizan en todas las fases de los protocolos y subprotocolos involucrados, bien para proteger datos del usuario, bien para proteger la negociación de otros parámetros de seguridad. Otro ejemplo que podríamos utilizar son los modos en los que puede operar el protocolo IKE dentro de IPsec para negociar los parámetros que regirán las fases posteriores de IPsec: aunque estos modos únicamente se utilizan en la fase inicial de IKE y al renegociar las claves criptográficas, el correcto desarrollo de la negociación de claves criptográficas depende completamente de la correcta utilización de estos modos.

Por último, el hecho de que las dos partes que participan en un protocolo o arquitectura de seguridad deban *utilizar los mismos parámetros de seguridad y comunicaciones* para poder proceder al intercambio y protección de la información de forma eficiente y con los niveles de seguridad deseados hace que todos aquellos parámetros que se pueden considerar opcionales o de configuración deban ser analizados y estudiados. En estos casos la importancia no sólo radica en la utilización de estos parámetros por sí mismos, sino también en los mecanismos con los que la implementación anuncia que procederá a utilizar la configuración o herramienta opcional, y la gestión que realiza de las respuestas (tanto de aceptación como de rechazo) que se recibe. Un ejemplo de parámetro de este tipo es la utilización de compresión en la capa IP utilizando IPComp ([140]) o el uso de certificados de cliente en SSL para autenticar al cliente ([58], [49]).

Teniendo en cuenta estos hechos, y partiendo de las especificaciones de protocolos y arquitecturas de seguridad en concreto, tales como el protocolo TLS o la arquitectura IPsec, podemos estudiar cuáles son los aspectos que son susceptibles de ser fuente de problemas al ser trasladados a una implementación real. Un estudio de los documentos en los que se definen algunos de estos protocolos y arquitecturas ([49], [58], [91], [86], [87], [25], [85], [54], [136], [70], [71], [133]) nos ha permitido identificar los aspectos que resultan críticos al evaluar el nivel de fidelidad con los que una implementación se ha ajustado a las especificaciones de un estándar.

Tanto los mecanismos criptográficos como los protocolos que conforman las arquitecturas de seguridad son los componentes fundamentales de estas arquitecturas, por lo que son utilizados intensivamente tanto durante el establecimiento de un canal seguro como en la posterior transmisión

de información a través del mismo. Además, tanto en los mecanismos criptográficos como en los protocolos utilizados el componente matemático - formal es muy elevado, lo que se complementa con la escasa flexibilidad que presentan a la hora de manejar estructuras de datos.

Complementariamente, los procesos de autenticación y gestión de claves que se encuentran en los protocolos y arquitecturas de seguridad son altamente dependientes tanto de los procesos criptográficos como de los protocolos utilizados para intercambiar la información: Sin embargo, la importancia de estos procesos de autenticación y gestión de claves, así como la entidad independiente de estos procesos dentro del establecimiento de canales seguros los convierte en otros mecanismos que deben ser analizados de cara a evaluar el nivel de conformidad de la implementación evaluada.

Por lo tanto, los aspectos que deberán ser estudiados son la **correcta implementación criptográfica**, el **desarrollo de los protocolos**, la **gestión de las claves criptográficas** y los **mecanismos de autenticación**. Adicionalmente, existen otros factores propios de cada fase de la arquitectura de seguridad y que también presentan características que hacen que necesiten ser evaluados, como se detallará a continuación.

3.2.1. Correcta implementación de los mecanismos criptográficos

Dado que muchos de los problemas y desviaciones del estándar se manifiestan en la implementación deficiente de los algoritmos criptográficos, es necesario que las metodologías evalúen dichas implementaciones, tanto en los algoritmos de cifrado como en los utilizados para calcular los resúmenes criptográficos (funciones *hash*), incluyendo en las pruebas y evaluaciones las herramientas criptográficas utilizadas: modos de cifrado, métodos de intercambio de claves (por ejemplo, aquí sería necesario probar los diferentes grupos de Diffie-Hellman definidos en el estándar, ...), métodos de autenticación y cualquier otra herramienta criptográfica que se utilice.

Todas estas pruebas implicarán un conjunto muy elevado de comprobaciones posibles a realizar, aunque posteriormente la cantidad de dichos tests que es necesario llevar a cabo no sea tan elevado, dado que el número de combinaciones $algoritmo\ de\ cifrado - tamaño\ de\ clave - tamaño\ de\ bloque - algoritmo\ de\ resumen - grupo\ de\ Diffie\ Hellman$ se reduce rápidamente en el caso de que no todos los algoritmos, tamaños de clave y bloque o grupos de Diffie Hellman están implementados o estén especificados como combinaciones válidas en la especificación de la arquitectura o protocolo de seguridad empleado: por ejemplo, en [49] se especifica que la única combinación de herramientas criptográficas que las implementaciones de TLS versión 1.1 deben soportar es `TLS_RSA_WITH_3DES_EDE_CBC_SHA`, mientras que las únicas combinaciones aceptables en una negociación son las que podemos

ver en la Tabla 3.1.

Tabla 3.1: Combinaciones de herramientas criptográficas válidas durante el proceso de negociación de TLS 1.1

Suite Criptográfica	**Interc. Clave**	
TLS_NULL_WITH_NULL_NULL	NULL	NULL - NULL
TLS_RSA_WITH_NULL_MD5	RSA	NULL - MD5
TLS_RSA_WITH_NULL_SHA	RSA	NULL - SHA
TLS_RSA_WITH_RC4_128_MD5	RSA	RC4_128 - MD5
TLS_RSA_WITH_RC4_128_SHA	RSA	RC4_128 - SHA
TLS_RSA_WITH_IDEA_CBC_SHA	RSA	IDEA_CBC - SHA
TLS_RSA_WITH_DES_CBC_SHA	RSA	DES_CBC - SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA	RSA	3DES_EDE_CBC - SHA
TLS_DH_DSS_WITH_DES_CBC_SHA	DH_DSS	DES_CBC - SHA
TLS_DH_DSS_WITH_3DES_EDE_CBC_SHA	DH_DSS	3DES_EDE_CBC - SHA
TLS_DH_RSA_WITH_DES_CBC_SHA	DH_RSA	DES_CBC - SHA
TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA	DH_RSA	3DES_EDE_CBC - SHA
TLS_DHE_DSS_WITH_DES_CBC_SHA	DHE_DSS	DES_CBC - SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	DHE_DSS	3DES_EDE_CBC - SHA
TLS_DHE_RSA_WITH_DES_CBC_SHA	DHE_RSA	DES_CBC - SHA
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	DHE_RSA	3DES_EDE_CBC - SHA
TLS_DH_anon_WITH_RC4_128_MD5	DH_anon	RC4_128 - MD5
TLS_DH_anon_WITH_DES_CBC_SHA	DH_anon	DES_CBC - SHA
TLS_DH_anon_WITH_3DES_EDE_CBC_SHA	DH_anon	3DES_EDE_CBC - SHA

Como podemos ver, teniendo 7 métodos de intercambio de clave, 5 cifradores diferentes y 3 funciones resúmenes diferentes utilizadas, únicamente 19 combinaciones de las 105 posibles pueden ser utilizadas en TLS.

3.2.2. Conformidad de protocolos y subprotocolos

Las ejecuciones de los protocolos y subprotocolos presentan otro de los aspectos que es necesario evaluar para llevar a cabo una evaluación del nivel de conformidad de una implementación de un protocolo o arquitectura de seguridad con el estándar. Esta evaluación deberá evaluar en qué medida la implementación analizada es capaz de proteger la información utilizando los diferentes protocolos y subprotocolos definidos en los estándares. Por lo tanto, es necesario considerar la máquina de estados que representa la ejecución de cada protocolo para evaluar si la máquina de estados de la implementación se corresponde con la descrita en los estándares.

En aquellos casos en los que el protocolo o arquitectura de seguridad

se componga de múltiples protocolos, como es el caso de IPsec, que consta de IKE para gestionar las claves criptográficas, y ESP y AH para proteger el tráfico), es necesario que cada protocolo se evalúe por separado (evaluación de la corrección de la implementación del protocolo), pero también será necesario comprobar tanto si el resultado de la ejecución del protocolo es el deseado (por ejemplo, si los parámetros de seguridad negociados con IKE son los esperados, especialmente cuando existen más de una propuesta de parámetros) como si la negociación y los parámetros negociados con dicho protocolo son aplicados correctamente durante el establecimiento de los túneles criptográficos.

3.2.3. Mecanismos de autenticación

Aunque estos mecanismos pueden ser incluidos en el área de *correcta implementación de las herramientas criptográficas* debido a que los mecanismos de autenticación se basan en herramientas criptográficas más o menos complejas, la importancia de la autenticación en los protocolos de seguridad aconseja prestar especial atención a este aspecto. Los diferentes mecanismos de autenticación que se utilizan en la actualidad pueden variar desde una simple contraseña (un secreto preestablecido) hasta cadenas de certificación que utilizan los servicios de una autoridad de certificación y una infraestructura de clave pública completa[1].

En el caso en que un protocolo soporte múltiples métodos de autenticación, la metodología de evaluación deberá proporcionar los medios para probar y evaluar todos ellos. Además, cuando sea posible utilizarlos configurando el nivel de seguridad (por ejemplo, en IKE es posible configurar si se desea verificar la validez del certificado X.509 que se recibe de la entidad con la que se establece el túnel criptográfico o si basta únicamente con que dicho certificado sea sintácticamente correcto), la metodología deberá evaluar los diferentes niveles que se hayan establecido como válidos en la especificación del protocolo en cuestión.

3.2.4. Opciones de gestión de las claves criptográficas

La mayoría de los protocolos de seguridad incluyen parámetros configurables (en el cliente, en el servidor o en ambos) en aspectos tales como cuándo deben caducar las claves secretas negociadas (expresado en tiempo pasado desde que se empezó a utilizar y/o en el volumen de datos que ha sido protegido con esa clave), y proporcionan métodos y técnicas para que

[1] En algunos casos se da la situación de que un mecanismo de autenticación se encuentra encubierto, por lo que puede ser incorrectamente identificado como un mecanismo nuevo (como es el caso de, por ejemplo, port-knocking ([46], [94])). Esto hace que se deba prestar atención a los usos que se hace de los mecanismos de autenticación para evitar cometer errores pasados, como se puede ver en ([79])

se pueda forzar una renovación de claves, etc.... Dada la importancia de estas medidas ([137], [6]), es necesario que la gestión de las claves se realice de la forma más segura posible, motivo por el que la gestión propuesta en los estándares ha sido sometida al escrutinio de la comunidad científica.

Por lo tanto, resulta necesario comprobar que las implementaciones de los protocolos y arquitecturas de seguridad estudiados llevan a cabo la gestión de las claves criptográficas de la forma descrita en su correspondiente especificación. Al igual que ocurría con los mecanismos de autenticación, los procedimientos de gestión de claves podrían englobarse en el apartado de implementación de las herramientas criptográficas, pero la importancia de esta gestión en la seguridad final del túnel criptográfico hace que se requiera de un análisis específico acerca de los aspectos de renovación y caducidad de las claves, tanto planificados (es decir, los acordados durante la negociación de los parámetros de seguridad) como no planificados (aquellos que son requeridos por alguna de las partes sin estar sujetos a los parámetros acordados en la fase de negociación.

3.2.5. Otras características

Normalmente las características definidas como opcionales en los estándares son una fuente de problemas, ya que la implementación que cada fabricante hace de ellas es muy dependiente de su modelo de desarrollo y de la habilidad al trasladar los diseños a una implementación concreta en un lenguaje de programación. Además, es habitual que las características que se definen como opcionales en una versión del estándar pasen a ser obligatorias en la siguiente versión (como es el caso de traversal-NAT ([92]), opcional en la primera versión de IPsec pero que debe ser soportado por todas las partes según la versión estandarizada en Diciembre de 2.005). Sin embargo, dado el carácter opcional, el juego de pruebas a realizar no será tan intenso y exhaustivo como lo debe ser al evaluar las implementaciones criptográficas o de ejecución del protocolo.

3.3. Métodos de validación de los factores de conformidad

Al llevar a cabo una validación de los factores anteriormente descritos como clave para poder asegurar la conformidad con el estándar de un protocolo o arquitectura de seguridad y, consiguientemente, la interoperabilidad con otras implementaciones de dicho protocolo o arquitectura, es necesario llevar a cabo un análisis de cada uno de dichos factores, para así estudiar las características de dicho elemento en la arquitectura de seguridad y el control del que disponemos de dicho factor en la implementación analiza-

da. Este análisis nos permitirá evitar el uso de técnicas de evaluación que resultan inviables o inadecuadas para el tipo de característica analizada.

3.3.1. Correcta implementación criptográfica

La evaluación de la correcta implementación de herramientas criptográficas es un tema que se ha tenido en cuenta por los diseñadores de estas herramientas, que en los diseños incluyen los llamados "vectores de prueba" ([137]). Estos vectores de prueba son conjuntos de entradas-salidas a la herramienta criptográfica pensados para evaluar la implementación, probando los valores límites para las entradas y para las operaciones internas que lleva a cabo la herramienta. Esto implica que para poder realizar las pruebas de los vectores de prueba es necesario poder controlar totalmente los valores de entrada a la herramienta criptográfica: claves, datos de entrada, generadores aleatorios, etc....

Sin embargo, al afrontar una evaluación de una implementación de IPsec como una caja negra (y más si el análisis se lleva a cabo de forma remota), nos encontramos con que no disponemos del control necesario sobre las herramientas criptográficas para poder llevar a cabo ese análisis. En la mayor parte de las implementaciones de IPsec las librerías criptográficas aparecen encapsuladas y el usuario final no dispone de acceso a dichas librerías para poder llevar a cabo las pruebas necesarias. Además, no es posible controlar aspectos como los generadores aleatorios que construyen las claves criptográficas de IPsec, o incluso el hecho de que existan diferentes librerías criptográficas para cada fase de IPsec. Disponer de dicho conocimiento convertiría el análisis en un estudio de caja blanca, que no es el objetivo de la metodología.

Por lo tanto, es necesario disponer de un método de evaluación de las herramientas criptográficas que pueda ser llevado a cabo por un examinador externo a la implementación, y sin disponer de conocimiento alguno acerca de la organización interna de la implementación. Además, para poder utilizar los resultados obtenidos como medida de interoperabilidad, es necesario que la evaluación sea tan parecida como sea posible a lo que una tercera parte se encontraría al establecer túneles criptográficos mediante IPsec con la implementación analizada.

Dado que los vectores de prueba son las únicas herramientas de validación práctica de las que disponemos a partir de los análisis formales del diseño de las herramientas, el método de evaluación escogido deberá ser capaz de utilizar, de alguna forma, estos vectores de pruebas para pronunciarse acerca de la implementación de las herramientas criptográficas en la implementación bajo nuestro estudio.

Tras analizar todos estos requisitos y limitaciones, el método de evaluación escogido consistirá en **aplicar la transitividad** para poder evaluar las

herramientas criptográficas: Al llevar a cabo el análisis de la implementación que se desea estudiar será necesario disponer de una implementación propia del protocolo o arquitectura de seguridad con la que negociar y establecer túneles criptográficos. Esta implementación se encontrará bajo nuestro control, y por lo tanto es factible validar que las herramientas criptográficas utilizadas en esta implementación "evaluadora" ofrecen los resultados esperados cuando se les somete a los vectores de prueba.

Una vez evaluada nuestra implementación, la forma de evaluar la implementación que no controlamos pasa por establecer túneles criptográficos entre ambas implementaciones, y procediendo al envío de mensajes de alto nivel a través de los túneles. En el caso de que la implementación de las herramientas criptográficas en la implementación bajo análisis presente alguna deficiencia, durante el intercambio de mensajes esta deficiencia saldrá a relucir en forma de mensaje interpretado incorrectamente o mensajes rechazados. Por lo tanto, los mensajes de alto nivel deberán ser escogidos cuidadosamente, para poder cubrir todas las opciones posibles de cifrado y descifrado, cálculo y validación de resúmenes, etc...

Al analizar qué tipo de mensajes de alto nivel se utilizarán para crear el tráfico, las posibilidades que se nos presentan son múltiples: desde tráfico de protocolos ampliamente utilizados, como HTTP, POP, DNS, ... hasta la utilización de generadores de tráfico propios. Sin embargo, estos protocolos de nivel de aplicación tienen el problema de que, a priori, no es posible conocer la respuesta que deberíamos obtener a una petición. Además, la dependencia de un servidor del protocolo escogido y de su correcto funcionamiento hacen que estas opciones no sean las más recomendables.

Sin embargo, mediante la utilización de mensajes ICMP ([130]) es posible conocer de antemano qué respuesta será la que debamos recibir. Por ejemplo: utilizando mensajes ICMP "echo request" con un campo de datos aleatorio (campo de datos que deberá aparecer igual en el mensaje de respuesta "echo reply") podremos comprobar si la implementación de IPsec que se está analizando ha sido capaz de descifrar y comprobar el resumen del mensaje recibido y cifrar correctamente y calcular el resumen del mensaje que debía devolver. La posibilidad de enviar otro tipo de mensajes ICMP, con sus consiguientes respuestas, amplía las posibilidades de cara a la evaluación de las herramientas criptográficas, ya que disponemos de información generada dinámicamente que la implementación de IPsec deberá procesar a través de los aparatos criptográficos que estemos evaluando en cada momento.

Por lo tanto, el método propuesto para la evaluación de las herramientas criptográficas de la implementación consiste en el establecimiento de túneles criptográficos protegidos mediante los protocolos ESP o AH (dependiendo de la herramienta criptográfica que deseemos evaluar) por los que se enviarán mensajes ICMP de los que se conoce de antemano la respuesta que

deberá generar la implementación analizada.

3.3.2. Conformidad de protocolos y subprotocolos

En la actualidad, y como se ha visto en la revisión de los protocolos y arquitecturas de seguridad llevado a cabo en el capítulo 2, los protocolos y arquitecturas de seguridad utilizados en la actualidad comprenden un conjunto de protocolos que se encargan de tareas que van desde la gestión de claves hasta la protección del tráfico entre los dispositivos. Cada uno de estos protocolos se encuentra altamente integrado con el resto de la arquitectura, por lo que las interdependencias son muy fuertes. Por ejemplo, de la negociación que se lleve a cabo con un protocolo dependerán los algoritmos de cifrado que se utilizarán en otro y cada cuánto habrá que renovar las claves de cifrado que se utilizan en dicho protocolo.

Por este motivo, la validación del desarrollo de los protocolos deberá realizarse no sólo de cada protocolo en concreto, sino que también deberá comprobarse que la interacción entre los diferentes protocolos es la esperada. Por este motivo la evaluación del correcto desarrollo de los protocolos deberá llevarse a cabo analizando cada uno de los protocolos individualmente primero, y junto al resto de protocolos después. Aunque este método implica un mayor número de pruebas, las implicaciones de un error en la interacción entre protocolos son demasiado grandes como para evitar llevar a cabo estos análisis.

En cuanto al análisis de cada uno de los protocolos, en general podemos encontrarnos dos tipos de protocolos: aquellos destinados a proteger la información (por ejemplo, ESP en IPsec) y aquellos encargados de gestionar los parámetros que gestionan esa protección de la información (como el protocolo de negociación en TLS, `Handshake protocol`). Dado que la funcionalidad y los objetivos de cada uno de estos tipos de protocolos son completamente diferentes también la evaluación deberá llevarse a cabo de forma diferente.

Para los protocolos destinados a proteger la información, el análisis de la conformidad con el estándar está descrito en el apartado "*Requisitos de Conformidad*" del estándar de cada protocolo[2]. En estos capítulos se declara que los requisitos que una implementación de esos protocolos debe cumplir para poder asegurar la conformidad con el estándar son:

- Interpretar cada campo de los mensajes generados y recibidos como se especifica en el estándar.
- Generar los mensajes con la sintaxis especificada.

[2]En aquellos protocolos y arquitecturas de seguridad en los que no se incluyó dicha sección en el estándar en su día, posteriormente han aparecido actualizaciones en las que se especifican cuáles son esos parámetros.

Por lo tanto, para evaluar si una implementación es conforme a los estándares necesitaremos evaluar si los mensajes que genera tienen el formato y la información especificada en el estándar, y si la implementación es capaz de interpretar mensajes que hayan sido generados por otros sistemas y que sean correctos según la especificación del estándar.

Hay que tener en cuenta que existen casos en los que protocolos o subprotocolos que son alternativos (es decir, es necesario escoger entre uno u otro), no presentan las mismas posibilidades de operación, por lo que la cantidad y tipo de pruebas que es necesario llevar a cabo en cada caso es diferente. Por ejemplo, en IPsec AH únicamente se admite un modo de operación mientras que ESP puede operar ofreciendo sólo confidencialidad, confidencialidad e integridad o confidencialidad, integridad y autenticación de origen. Esto quiere decir que todos los modos de operación de ESP deberán ser tratados de forma independiente y deben evaluarse con su propia batería de pruebas.

En lo tocante a los protocolos de gestión de claves su evaluación requiere de un análisis más elaborado. Partiendo también de los "*Requisitos de Conformidad*" especificados en los documentos de los estándares, podemos identificar cuáles son las características que es necesario evaluar para llevar a cabo el análisis de conformidad del protocolo. En concreto, estos aspectos (que se reflejarán en más o menos pruebas dependiendo del desarrollo interno del protocolo concreto que está siendo evaluado) son:

- Desarrollar un proceso completo de negociación e intercambio de claves, utilizando todos los modos de operación en los que el protocolo pueda trabajar.
- Desarrollar un proceso completo de negociación de parámetros de protección de la información, utilizando todos los modos de operación en los que el protocolo pueda trabajar.
- Llevar a cabo procesos de autenticación utilizando los diferentes mecanismos soportados por el protocolo.

Adicionalmente, otras características que no se encuentren explícitamente englobadas en los puntos anteriores, pero que se encuentren relacionadas con el desarrollo de los protocolos de gestión de claves deberá ser evaluado. Por ejemplo, en IPsec la característica de "*Perfect Forward Secrecy*" representa un cambio sustancial tanto en la seguridad del sistema como en las propiedades de la información intercambiada, por lo que deberá ser incluida en en conjunto de factores a analizar. [3].

[3]Sin embargo, como veremos al desarrollar la metodología para la validación de la conformidad y evaluación del rendimiento en implementaciones de IPsec en el capítulo 5,

Por otro lado, la autenticación será estudiada en detalle más adelante, por lo que en la metodología no se incluirán los diferentes métodos de autenticación en la evaluación de los protocolos en sí mismos. De esta manera podremos diferenciar los problemas que puedan derivarse de una ejecución incorrecta de los protocolos de aquellos cuyo origen es el procesado de autenticación

Como último paso en la evaluación de la ejecución de los protocolos y subprotocolos implicados, es necesario incluir en el diseño de las pruebas de conformidad aquellas características que, sin pertenecer estrictamente hablando al desarrollo de los protocolos, afectan al modo en el que se lleva a cabo la ejecución de los mismos durante la protección de tráfico. Por ejemplo, la posibilidad que ofrece IPsec de separar en túneles criptográficos diferentes aquellas comunicaciones con diferentes protocolos, servicios, etc... hace que la máquina de estados interna se ejecute de forma diferente cuando esta característica se lleva a cabo y cuando no. Por este motivo, la correcta ejecución de los protocolos involucrados debe evaluarse teniendo en cuenta estos factores.

Para llevar a cabo esta evaluación se debe crear tráfico desde la red protegida por la implementación objeto de nuestro análisis, en el que únicamente cambie un parámetro de las características de comunicaciones (puerto, protocolo, equipo origen, equipo destino, etc...), y comprobar cómo la implementación evaluada solicita la creación de nuevos túneles criptográficos. El caso contrario, en el que el tráfico se origina hacia la implementación que está siendo analizada no debería presentar problemas, ya que la implementación únicamente recibirá solicitudes para establecer nuevos túneles criptográficos y responderá a ellos; posteriormente el tráfico de vuelta será enviado por un túnel u otro en función de la base de datos de políticas, lo que resulta en un caso similar a la generación de tráfico desde la red protegida, como proponíamos anteriormente.

Un aspecto importante de esta prueba es que el establecimiento de nuevos túneles criptográficos es una operación costosa en recursos (como se ha verá en la sección 4.2). Por lo tanto, el tráfico que se genere para comprobar si la implementación es capaz de aislar el tráfico en túneles independientes deberá espaciarse en el tiempo lo suficiente para permitir que la implementación negocie todos los parámetros de seguridad para cada túnel. Adicionalmente, no se deberá enviar tráfico por el túnel, ya que con el establecimiento de la conexión es suficiente para forzar el establecimiento de un nuevo túnel, y la presencia de tráfico que debe ser cifrado sólo empeoraría el rendimiento de la implementación en la generación de túneles.

Por todo lo anterior podemos decir que el establecimiento de conexiones

no es obligatorio para las implementaciones de IPsec el incluir "Perfect Forward Secrecy", por lo que estas posibles situaciones deben ser tenidas en cuenta a la hora de desarrollar metodologías y baterías de pruebas

TCP desde la red que protege la implementación objeto de la evaluación, **sin envío de tráfico posterior** es el método más adecuado para evaluar la posibilidad de aislamiento del tráfico en diferentes túneles criptográficos.

Finalmente, a modo de evaluación de la interoperabilidad de todos los protocolos y subprotocolos de la arquitectura o protocolo de seguridad trabajando conjuntamente, es necesario incluir la evaluación del proceso completo de protección de la información, desde el establecimiento de túneles criptográficos hasta la transmisión de información a través de dichos túneles, como forma de evaluar la correcta interacción de todos los protocolos.

3.3.3. Autenticación

Como hemos comentado anteriormente, uno de los requisitos para poder afirmar que una implementación de un protocolo o arquitectura de seguridad es conforme al estándar es poder autenticar y autenticarse al establecer un túnel criptográfico. El proceso de autenticación aparece definido en los estándares como parte de los protocolos (ya que es un paso fundamental en el desarrollo de la negociación inicial para establecer nuevos túneles criptográficos, y también para validar el origen de los mensajes protegidos mediante los protocolos correspondientes), por lo que su estudio podría incluirse en el análisis de la ejecución de los subprotocolos implicados.

Igualmente, debido al fuerte uso de herramientas criptográficas en los diferentes métodos de autenticación soportados por la arquitectura de seguridad, sería también posible haber llevado a cabo el estudio de la autenticación junto con el resto de algoritmos y mecanismos criptográficos.

Sin embargo, debido a que la autenticación debe hacer uso de las herramientas criptográficas y de los protocolos y subprotocolos que conforman la arquitectura, es necesario independizar las pruebas de estas herramientas de las de la autenticación, para evitar identificar incorrectamente problemas en la ejecución del protocolo (o en la utilización de determinado algoritmo criptográfico) como problemas de la autenticación, o viceversa.

Es importante también tener en cuenta que, de cara a la conformidad con el protocolo, es necesario evaluar cuáles de los métodos de autenticación incorporados a las implementaciones se encuentran recogidos por el estándar como necesarios, y cuáles son únicamente opcionales. En algunos casos es habitual encontrar métodos de autenticación opcionales que se encuentran ampliamente extendidos y aceptados, u otros casos en los que recientes modificaciones al estándar han incorporado nuevos métodos de autenticación (normalmente, con carácter opcional), pero que por su implicación en la seguridad o interoperatividad del protocolo o arquitectura de seguridad con otros mecanismos y herramientas de autenticación, es altamente recomendable estudiar la disponibilidad de dichos mecanismos en las implementaciones que se evalúen. Un ejemplo de este tipo de mecanismos es la autenticación

mediante EAP ([1]), que permite la integración de servicios de autenticación que hagan uso de diferentes protocolos particulares para llevar a cabo dicha autenticación, abstrayendo el uso y manejo concreto de las credenciales que se utilicen en cada caso, tal como se expone en ([131]).

La evaluación de la implementación de los mecanismos de autenticación deberá llevarse a cabo haciendo que la implementación ejerza los roles de parte autenticada y de autenticador, de forma que sea posible evaluar los mecanismos tanto de generación como de validación de credenciales. Este proceso se llevará a cabo con todos los métodos de autenticación soportados en la especificación del protocolo o arquitectura de seguridad que implemente el dispositivo o software bajo análisis, aunque siempre teniendo en cuenta cuáles son los métodos de autenticación exigidos por la especificación, y cuáles son opcionales.

3.3.4. Gestión de Claves

Dentro de todo el proceso de gestión de claves que llevan a cabo los protocolos y arquitecturas de seguridad, uno de los aspectos que requieren especial atención es la gestión de las claves criptográficas negociadas. Esta gestión es la que se encarga de renegociar las claves criptográficas para evitar un uso excesivo de la misma clave y de proporcionar mecanismos para aislar las claves de diferentes fases de la arquitectura, de forma que el compromiso de una clave no afecta al resto de claves negociadas (por ejemplo, en IPsec dicha propiedad es cierta si se utiliza la opción de *Perfect Forward Secrecy*, PFS).

Al evaluar la gestión de las claves, es necesario comprobar dos aspectos diferentes: Por un lado, hay que comprobar que la implementación estudiada es capaz de detectar la caducidad de una clave e iniciar el proceso de renovación de la misma de acuerdo a lo definido en la especificación del protocolo. Esta renovación por caducidad puede dispararse por diversos factores, tales como el tiempo que dicha clave ha estado en uso, o la cantidad de información que se ha protegido con ella. Además, los sistemas implicados en el túnel criptográfico pueden contar con algún mecanismo de sincronización o no, por lo que dicho aviso de caducidad de la clave puede llegar de forma síncrona o asíncrona, lo que hace necesario evaluar el comportamiento de la implementación tanto cuando actúa como el sistema que avisa de que la clave ha caducado, como cuando su papel es el de receptor de dicha notificación.

Por otro lado, la respuesta de la implementación ante solicitudes asíncronas de renegociación de la clave no correspondientes con caducidades planificadas de la clave es el otro aspecto de la gestión de las claves criptográficas que debe ser validado. En este aspecto es necesario comprobar cuál es el comportamiento de la implementación objeto del análisis a la hora de solicitar una renegociación de claves, así como su reacción cuando recibe

una solicitud similar.

Al igual que se hacia hincapié a la hora de validar los mecanismos de autenticación, de cara a establecer el nivel de conformidad de una implementación con el estándar del protocolo o arquitectura de seguridad que desarrolla es necesario diferenciar entre los comportamientos definidos como válidos en la especificación de dicho protocolo o arquitectura, y aquellos comportamientos que, incluso estando ampliamente extendidos entre las implementaciones más populares, únicamente aparecen recogidos como opcionales. Al igual que se comentaba anteriormente, aquellas opciones, mecanismos y comportamientos que resulten de especial interés para la protección de la información o para la funcionalidad del sistema podrán ser analizadas, pero siempre teniendo en cuenta su carácter opcional.

3.3.5. Otras características

Como colofón al estudio de la conformidad con el estándar, es necesario evaluar en qué medida las implementaciones de un protocolo o arquitectura de seguridad hacen uso de otras herramientas que, si bien están definidas en el estándar, su utilización es bien opcional, bien enfocada a determinadas situaciones o arquitecturas específicas. Estas herramientas puede ofrecer funcionalidad adicional a la incluida en el protocolo o arquitectura, o pueden habilitar el uso de características de seguridad en determinadas arquitecturas, topologías de red, o situaciones particulares.

Para cada una de estas características será necesario definir un escenario de pruebas en el que se den las condiciones necesarias para que la implementación del protocolo de seguridad tenga que hacer uso (tanto por iniciativa propia como a petición del otro extremo del túnel criptográfico) de estas opciones. Por ejemplo, para poder validar la conformidad con el estándar del NAT traversal en una implementación IPsec será necesario utilizar una arquitectura de red que permita a la implementación evaluada hacer uso de la tecnología de NAT, y así poder generar tráfico que deba ser protegido mediante IPsec utilizando NAT traversal.

3.4. Consideraciones de implementación

Como se ha podido deducir de los métodos propuestos para analizar los aspectos de conformidad con el estándar, para llevar a cabo todas las pruebas es necesario poder enviar y recibir mensajes de negociación de claves y parámetros de seguridad, de caducidad de las mismas claves, etc.... Por lo tanto, se debe disponer de una implementación de IPsec que podamos controlar y a la que podamos forzar a enviar mensajes incorrectos cuando sea necesario. Además, esta implementación debe ofrecernos la seguridad

de que, en el caso de que aparezcan problemas al establecer los túneles criptográficos, todos los problemas serán debidos a la implementación que está siendo analizada, y no a la que estamos utilizando para generar las conexiones, etc...

Mientras que es importante disponer de una implementación que podamos controlar según nuestras necesidades (existen múltiples implementaciones de código abierto que nos permitirían llevar a cabo esta tarea), el hecho de que la implementación utilizada deba ser conforme a los estándares nos introduce en el problema de cómo validar si la implementación es conforme o no. De hecho, el problema se presenta realmente al evaluar el correcto desarrollo de los protocolos, ya que para las herramientas criptográficas podemos usar los vectores de pruebas con las librerías criptográficas que se utilicen, de forma que podamos validar dichas librerías. Sin embargo, validar la ejecución del protocolo es un problema sin una solución tan clara.

Por un lado se podría utilizar una implementación de código abierto y llevar a cabo una auditoría del código para verificar si el comportamiento es correcto o no, pero este método presenta los mismos problemas que se comentaron para los métodos de validación del desarrollo en la sección 2.2. Por lo tanto, este método debería ser evitado en la medida de lo posible.

Por otro lado, podríamos desarrollar nuestra propia implementación del protocolo o arquitectura de seguridad correspondiente y someterla a un proceso de evaluación estableciendo túneles criptográficos con múltiples implementaciones diferentes. A medida que surjan problemas se deberá estudiar a qué implementación de las implicadas en el túnel criptográfico se deben dichos problemas. Este método presenta la desventaja de que, al estudiar una implementación que no ha participado del proceso de depuración de nuestra plataforma, no es posible determinar a priori si la presencia de problemas de interoperatividad se debe a la implementación que está siendo evaluada o a la que se está utilizando para evaluar al resto.

Otra posible solución pasaría por la utilización de implementaciones estandarizadas de los protocolos o arquitecturas de seguridad. En muchos casos, al tiempo que se estandariza un protocolo o arquitectura de comunicaciones el organismo internacional correspondiente publica implementaciones de ese protocolo o arquitectura de forma que estén disponibles para el público en general. En otros casos, como resultados de proyectos de investigación o de labores de evaluación en centros públicos, se pone a disposición de la comunidad internacional una implementación que, sin ser oficial, sí que es conforme con lo estandarizado en los documentos correspondientes. En estos casos en los que estas implementaciones se encuentran disponibles, sería posible utilizarlas como punto de partida para validar la conformidad de la implementación que nos interesa.

En otro orden de cosas, es necesario tener en cuenta que para llevar

a cabo todos los análisis que se describirán en la memoria, será necesario disponer de una infraestructura de red que nos permita enviar tráfico a través de los túneles criptográficos que se establezcan. Por lo tanto, debe asegurarse el acceso correspondiente a la implementación y su configuración a través de la red (es decir, confirmando que no hay cortafuegos bloqueando el tráfico, que la configuración de los túneles criptográficos no previene el envío de la configuración, etc....). Idealmente, se contaría con una infraestructura dedicada para llevar a cabo las pruebas en cuestión, lo que mejoraría la ejecución de los diferentes tests y evitaría posibles falsos informes de error.

Otra consideración que hay que realizar es que es necesario disponer de acceso a la configuración de la implementación para poder cambiarla a medida que la realización de las oportunas pruebas así lo requiera. Igualmente, será obligatorio gestionar equipos que se encuentren en la red protegida por la implementación evaluada, para poder generar tráfico hacia el túnel criptográfico cuando se requiera.

Por último, es preciso destacar que el análisis de la conformidad de una implementación de un protocolo o arquitectura de seguridad presenta el problema del orden en el que deberían evaluarse los diferentes factores: Si se comienza evaluando la conformidad criptográfica nos encontramos con que para probar la conformidad de los algoritmos utilizados debemos llevar a cabo un desarrollo, completo o parcial, de los protocolos de negociación y/o protección de la información. Sin embargo, si se comenzase evaluando la correcta implementación de los protocolos nos encontraríamos con que las herramientas criptográficas son utilizadas desde el primer momento para proceder con las negociaciones.

Esto nos introduce en un problema circular de dependencias, con difícil solución práctica. En esta tesis se ha optado por comenzar evaluando la conformidad criptográfica, ya que de esta forma podremos descartar muchos de los problemas que se presentan en las implementaciones de hoy en día. Además, la presencia de errores en la ejecución de alguno de los protocolos se manifestará en forma del mismo error, independientemente de las herramientas criptográficas utilizadas, lo que no ocurriría si el problema es sólo de una herramienta criptográfica. Mediante esta técnica estamos en disposición de evaluar la conformidad criptográfica y el desarrollo de los protocolos de forma muy fiable, allanando el camino para las evaluaciones que siguen a aquéllas. En cualquier caso, los errores detectados en esta fase deberán acompañarse de todos los datos que puedan ser recopilados, con el fin de proporcionar información suficiente para permitir detectar el error en caso de que así sea.

Capítulo 4

Análisis del Rendimiento

4.1. Introducción

En este capítulo se describirán cuáles son los aspectos del rendimiento que es importante conocer en las implementaciones de protocolos y arquitecturas de seguridad, comparando estos aspectos con aquellos recogidos en otros conjuntos de pruebas de rendimiento para dispositivos y software de red propuestos por la comunidad científica. Esté análisis de la particularidad de los protocolos de seguridad en general servirá de piedra angular para presentar a continuación los métodos y herramientas necesarios para llevar a cabo una medición de dichos aspectos de rendimiento, así como otras particularidades que deben ser tenidas en cuenta al realizar estas mediciones. Con el fin de poder referirnos genéricamente a los protocolos y arquitecturas de seguridad sin tener que utilizar todas las diferentes nomenclaturas existentes, en este capítulo se hablará de *asociación de seguridad* para referirnos al conjunto de parámetros negociados que definen cómo se protegerán las comunicaciones entre dos sistemas (concepto equivalente a las asociaciones de seguridad de IPsec y similar al de las sesiones SSL).

4.2. Identificación de los factores de rendimiento

Como ya se ha comentado en la sección 2.3.1, la evaluación del rendimiento de los protocolos de seguridad tiene su origen en la evaluación del rendimiento de los protocolos y redes de comunicaciones, compartiendo con esta rama las bases, tanto metodológicas como de sujetos del análisis, que conforman su estructura, aunque las diferencias entre los protocolos de comunicaciones con seguridad integrada y los protocolos sin ella hacen que ambas disciplinas se hayan alejado progresivamente en los últimos años. Por este motivo nos encontraremos con que en múltiples casos los parámetros que se intentan evaluar son similares a los que se tienen en cuenta en la eva-

luación tradicional de redes y protocolos, aunque los motivos para realizar las pruebas, así como los detalles de dichas pruebas, son muy diferentes.

4.2.1. Análisis de SSL

Dado que la principal diferencia entre un protocolo de comunicaciones "tradicional" y un protocolo de seguridad es la inclusión de técnicas criptográficas para proporcionar determinados servicios de seguridad en el protocolo de seguridad, comenzaremos nuestro análisis estudiando cuál es el impacto en el rendimiento al incluir estas herramientas criptográficas en un protocolo de comunicaciones en seguridad. Con este fin se ha desarrollado una prueba de concepto con un cliente y servidor de `echo` (el cliente envía un mensaje al servidor y éste devuelve el mismo mensaje) sobre TCP, y posteriormente se le han incluido características de seguridad utilizando el protocolo SSL (Secure Socket Layer). Los motivos para utilizar SSL en esta prueba de concepto son varios: Por un lado, para obtener información fiable acerca de los tiempos de ejecución y recursos empleados es necesario utilizar un "profiler", que se enlaza dinámicamente con el código a analizar y registra los recursos utilizados, grafos de ejecución, etc.... En el caso de otros protocolos y arquitecturas como IPsec, dado que las implementaciones se integran con el núcleo del sistema operativo para poder crear capas adicionales en la pila TCP/IP de forma transparente a aplicaciones y usuarios, el uso de uno de estos "profiler" representa una operación compleja (ya que obtendríamos información de todas las llamadas y funciones del núcleo del sistema operativo, entre las que habría que localizar las correspondientes a IPsec) y poco fiable, dado que por su propia naturaleza los profiler introducen retardos en la ejecución del código; si esos retardos se producen en el propio núcleo del Sistema Operativo, nos encontramos con que el rendimiento que estamos midiendo tiene poco o nada que ver con el rendimiento real del protocolo de seguridad.

Por otro lado, aunque los servicios de seguridad proporcionados por SSL e IPsec son, en muchos casos, similares (aunque aplicados a capas de la pila TCP/IP diferentes, o con herramientas criptográficas más o menos complejas y elaboradas), la complejidad de IPsec es mucho mayor. Esta elevada complejidad hace que al implementar una prueba de concepto que sirva de guía inicial a posteriores investigaciones se elija SSL, señalando en los resultados las posibles diferencias con las que nos podremos encontrar posteriormente, para evitar posibles confusiones.

Las características técnicas de los sistemas utilizados para llevar a cabo esta prueba de concepto aparecen recogidas en la Tabla 4.1 [1] [2]. Antes de

[1] Las opciones del compilador utilizadas para la activación y enlazado del profiler no aparecer recogidas en esta tabla

[2] La opción del compilador `-lgnutls` se utiliza únicamente para las versiones del cliente

proceder a analizar los resultados, hay que destacar dos aspectos importantes de esta prueba de concepto. El primero de ellos es el hecho de que la propia utilización de los "profiler" introduce variaciones en el rendimiento que los sistemas, ralentizando su ejecución debido a las medidas que constantemente se están tomando para proporcionar la información deseada. Por otro lado, también hay que definir el concepto de *"solicitud de memoria"* que aparecerá posteriormente como un dato proporcionado por uno de estos "profiler": "Solicitud de memoria" hace referencia a la cantidad de memoria que un programa solicita al sistema operativo a través de funciones como `malloc` en C. Por lo tanto, esta solicitud de memoria no hace referencia a la memoria real que necesita un programa para ejecutarse (ya que la memoria utilizada por las variables que no requieran de solicitud de memoria explícita por parte del programador no aparecen recogidas en el informe), ni al tamaño máximo del programa en memoria (ya que esas solicitudes de memoria serán liberadas en un momento u otro mediante instrucciones `free`), valor que aparece en esta memoria como "Memoria Residente" (tamaño máximo del programa en RAM) y "Memoria Virtual" (cantidad total de memoria reservada para el programa). Como veremos posteriormente, todos estos datos proporcionan información acerca del uso intensivo de la memoria por parte de las implementaciones de herramientas criptográficas muy valiosa.

Tabla 4.1: Especificaciones de los sistemas empleados en la prueba de concepto de SSL

Procesadores	Pentium M Centrino 800 MHz
Memoria	1024 MB en cada equipo
Red	Fast Ethernet con conexión directa entre cliente y servidor
Sistema Operativo	Gentoo Linux con kernel 2.6.15
Compilador	GCC 3.4.5
Opciones del Compilador	-Wall -O3 -lgnutls
Librerías Criptográficas	GnuTLS 1.2
Algoritmos Criptográficos	AES 256 & SHA1
Método de Autenticación	Certificados X.509
Profiler de CPU	FunctionCheck
Profilers de Memoria	ccmalloc y exmap

Los tiempos de ejecución obtenidos para el cliente se muestran en la Figura 4.2.1, mientras que los resultados del servidor pueden verse en la Figura 4.2.1. En ambos casos la serie nombrada `GnuTLS 1` se refiere a una implementación del canal seguro con SSL en el que cliente y servidor úni-

y del servidor que utilizan SSL

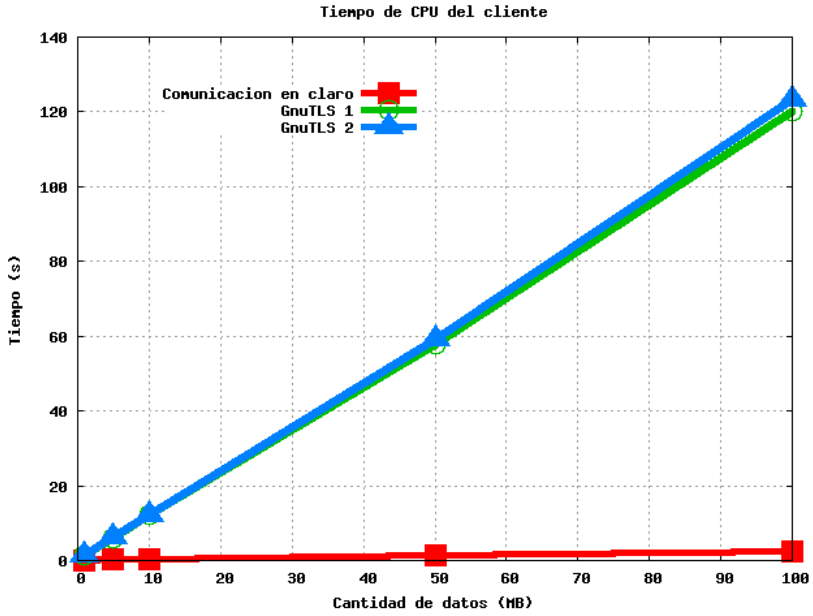

Figura 4.1: Tiempo empleado por el cliente en la trasmisión y recepción de una determinada cantidad de datos utilizando diferentes canales de comunicación

camente cifran los datos que se transmiten, mientras que en la implementación `GnuTLS 2` se incluye la autenticación tanto de cliente como de servidor. Como podemos observar, al utilizar el canal protegido con SSL el tiempo necesario para la transmisión de la misma cantidad de información se multiplica por más de 20 con respecto a la transmisión de la misma cantidad de información sin ningún tipo de protección. Adicionalmente, podemos apreciar cómo la implementación que lleva a cabo la autenticación del cliente y del servidor ofrece un rendimiento ligeramente peor. En el caso concreto de estas pruebas de concepto la diferencia es mínima, pero extrapolando esta información a IPsec, en el que periódicamente es necesario descartar la información criptográfica negociada y volver a acordar nuevas claves basándose en la información de autenticación, el impacto puede ser mucho mayor. Asimismo, es necesario considerar el impacto en entornos en los que múltiples clientes se autentican frente al mismo servidor, ya que lo que para el cliente es una única autenticación y puede no representar una sobrecarga considerable, en el servidor se convierte en un gran volumen de operaciones de autenticación, lo que sí que puede representar un impacto importante en el rendimiento. Estas situaciones son más propicias a aparecer en aquellos entornos en los que múltiples cliente se conectan al mismo servidor de seguridad en un corto espacio de tiempo, como puede ser el inicio de la jornada laboral en una empresa.

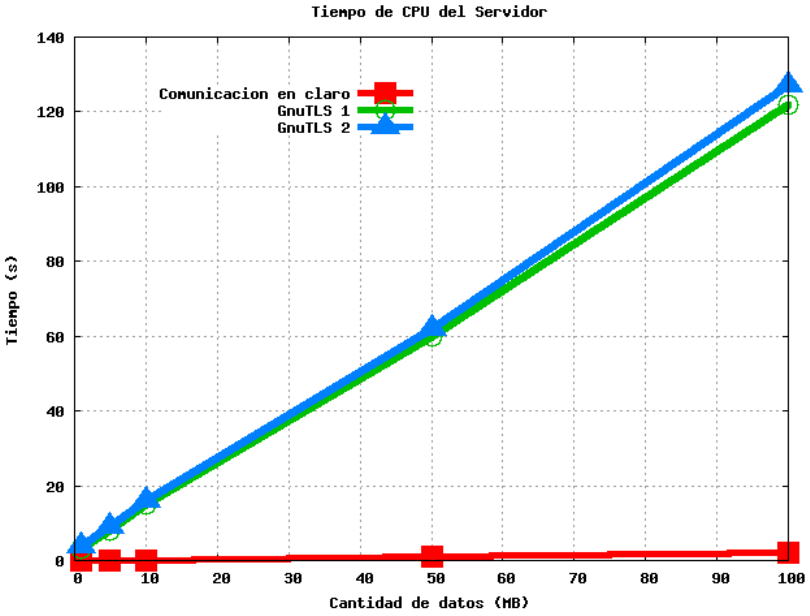

Figura 4.2: Tiempo empleado por el servidor en la transmisión y recepción de una determinada cantidad de datos utilizando diferentes canales de comunicación

En lo tocante a la utilización de la memoria, en la Tabla 4.2 podemos ver los resultados ofrecidos por `ccmalloc` para cada una de las implementaciones de cliente y servidor[3]. El efecto que estas solicitudes de memoria tienen en el rendimiento es muy importante, ya que, como primera conclusión que podemos extraer de estos resultados, la cantidad de memoria disponible en el sistema definirá qué tipo de autenticación es posible utilizar (ya que la cantidad de memoria necesaria para utilizar un secreto preestablecido (por ejemplo, una contraseña) no es la misma que la necesaria para utilizar un certificado de clave pública X.509 tanto en cliente como en el servidor), y, por extensión, cuáles son los requisitos mínimos para los diferentes dispositivos que deseen establecer un túnel criptográfico utilizando dichos métodos de autenticación.

Por otro lado, podemos ver también que la diferencia de solicitudes de memoria entre las diferentes implementaciones de SSL no representa la mayor parte de dichas solicitudes, por lo que podemos deducir que son las operaciones de protección de la información (es decir, el cifrado y el descifrado). Esto quiere decir que el cifrado y descifrado de bloques de datos no son únicamente operaciones costosas en tiempo de CPU, sino también

[3]Dado que las librerías de comunicaciones del sistema no fueron compiladas con los enlaces del "profiler" para obtener información acerca de ellas, `ccmalloc` devuelve un valor de 0 reservas de memoria para el servidor en claro.

Tabla 4.2: Solicitudes de memoria (en KBytes) durante la ejecución de la prueba de concepto

Cantidad de Datos	Cliente en claro	Cliente GnuTLS 1	Cliente GnuTLS 2
1 MB	1,4	10.011,3	10.171,1
5 MB	1,4	48.873,4	49.023,1
10 MB	1,4	97.448,6	97.609,0
50 MB	1,4	486.046,3	486.196,1
100 MB	1,4	971.786,4	971.918,2
	Servidor en claro	Servidor GnuTLS 1	Servidor GnuTLS 2
1 MB	0	10.200,7	10.588,4
5 MB	0	49.233.7	49.759,4
10 MB	0	97.852.6	98.107,9
50 MB	0	486.325.7	486.587,3
100 MB	0	972.441.8	972.517,9

en memoria, y por lo tanto, esto habrá de tenerse en cuenta al diseñar el conjunto de pruebas que evalúen en rendimiento del sistema. En concreto, será necesario evaluar cómo afecta la existencia de túneles criptográficos por los que se transmite información a la capacidad de establecimiento de nuevos túneles criptográficos, así como cuál es el impacto de variar la velocidad a la que se transmite información por esos túneles (y por ende, la velocidad a la que la implementación del protocolo de seguridad solicita memoria al sistema).

En cuanto a la memoria realmente utilizada por cada una de las implementaciones, en la Tabla 4.3 podemos ver el resumen de los datos obtenidos con la herramienta Exmap acerca de la memoria residente (es decir, la memoria reservada en la RAM física del equipo) y la memoria virtual (máximo tamaño de memoria que puede utilizar el proceso) para cada una de las implementaciones, tanto clientes como servidores. Dado que la cantidad de memoria reservada en un momento dado y en total por las implementaciones no varía con el tamaño de la información transmitida[4], en esta tabla podemos ver cómo únicamente disponemos de los dos datos ya comentados para cada implementación.

[4]Esto se debe a que al tener que solicitar una nueva porción de memoria se libera la anterior, aspecto que no quedaba reflejado si únicamente nos atenemos a los datos de la Tabla 4.2

Tabla 4.3: Tamaño de Memoria Residente y Virtual (en KBytes) de las pruebas de concepto

		Transferencia en claro	GnuTLS 1	GnuTLS 2
Cliente	Residente	77.8	428.4	534.2
	Virtual	1492	2676	2688
Servidor	Residente	63.0	426.6	529.8
	Virtual	1360	2672	2680

4.2.1.1. Predicción del rendimiento de las soluciones de seguridad

Uno de los motivos por los que es tan importante el conocer el rendimiento de nuestra solución de seguridad es que no es factible asegurar a priori el rendimiento que una determinada solución de seguridad va a poder proporcionar, incluso en sistemas similares, dado que el rendimiento de las operaciones criptográficas es variable. Factores como la optimización por hardware, la inclusión de múltiples núcleos en cada CPU, la optimización que haya introducido el compilador, otros procesos que se ejecuten en el sistema, etc. . . . Todos estos aspectos influyen en gran medida en el rendimiento que nuestro sistema pueda ofrecer, y dificultan sobremanera la estimación del servicio que puede ofrecer nuestro sistema, como puede verse en [104] y en [7]. En estos dos estudios se puede ver que el impacto de protocolos de seguridad (IPsec y SSL en el primero de ellos, y exclusivamente SSL en el segundo) en el rendimiento varía entre el 10 y el 90 %, según los algoritmos de cifrado y resumen que se utilicen, si es posible utilizar tarjetas criptográficas o implementación hardware de las herramientas criptográficas utilizadas, la diferencia entre el tamaño de bloque de los cifradores y el tamaño de la información a proteger, la existencia de redundancia o balanceo de carga en el servicio de seguridad y si se deciden reutilizar claves criptográficas o no.

Para llevar a cabo un estudio acerca del rendimiento en el cómputo de operaciones criptográficas y la viabilidad de la predicción de este rendimiento de forma acertada, se ha realizado un análisis comparativo utilizando varios equipos con diferentes familias de procesadores. En este caso, y debido al diseño de las pruebas llevadas a cabo, únicamente la información acerca del procesador es relevante, dado que nunca se llevan a cabo pruebas en paralelo ni se trabaja con más de 8 KBytes de datos simultáneamente. La lista de equipos involucrados en la evaluación puede verse en la Tabla 4.1.

Las pruebas llevadas a cabo han consistido en la realización de los test de velocidad incluidos en OpenSSL, que miden la cantidad de operaciones criptográficas que un procesador es capaz de realizar por unidad de tiempo,

Tabla 4.4: Equipos utilizados en la evaluación del rendimiento

Equipo 1	Intel Pentium II 350 MHz
Equipo 2	AMD Athlon K7 550 MHz
Equipo 3	Intel Pentium III 750 MHz
Equipo 4	Intel Pentium M (Centrino) 2.0 GHz
Equipo 5	Intel Pentium D (Dual Core) 2.8 GHz
Equipo 6	AMD Athlon64 3000+ 2.0 GHz
Equipo 7	Intel Pentium IV 3.6 GHz

Tabla 4.5: Comparación del rendimiento de varios equipos al cifrar y calcular resúmenes criptográficos, en miles de operaciones por segundo

	1	**2**	**3**	**4**	**5**	**6**	**7**
MD4	57.453	111.662	150.733	346.824	411.080	396.987	523.556
MD5	47.508	96.600	114.691	326.515	508.208	234.239	651.656
SHA1	18.569	58.580	62.300	206.982	282.564	190.068	361.643
RC4	35.821	76.149	82.706	311.886	114.466	157.284	143.746
Blowfish	11.859	23.399	26.206	74.069	103.244	78.731	132.831
AES128	7.264	14.317	16.512	45.385	44.828	99.555	84.574

ofreciendo información de esta velocidad para cada algoritmo de cifrado y firma disponible en el sistema.

En la Tabla 4.5 se encuentran reflejados los resultados obtenidos a la hora de evaluar el rendimiento de los equipos al realizar operaciones de cifrado sobre bloques de 8KBytes de datos, utilizando los algoritmos de cifrado RC4, Blowfish y AES128, y al calcular resúmenes criptográficos de esos bloques de datos con las funciones MD4, MD5 y SHA1. Esta misma información se puede encontrar en la Figura 4.3, para poder observar la evolución y comparación de los resultados de forma gráfica. Como podemos observar, la velocidad del procesador no es un indicativo válido del rendimiento que dicho procesador es capaz de ofrecer a la hora de cifrar o calcular resúmenes criptográficos de bloques de información.

Como podemos observar, estas pruebas ofrecen resultados sorprendentes, tanto en cuanto un procesador concreto aumenta de forma sustancial su rendimiento al utilizar un algoritmo de cifrado determinado (como es el caso del Pentium M (Centrino) al cifrar datos con RC4, o el AMD Athlon64 al utilizar AES128), mientras que otros procesadores reducen considerablemente su capacidad de proteger los datos al utilizar otros algoritmos (como es el caso del Pentium IV y el Pentium D (Dual Core) con RC4, o el Pentium

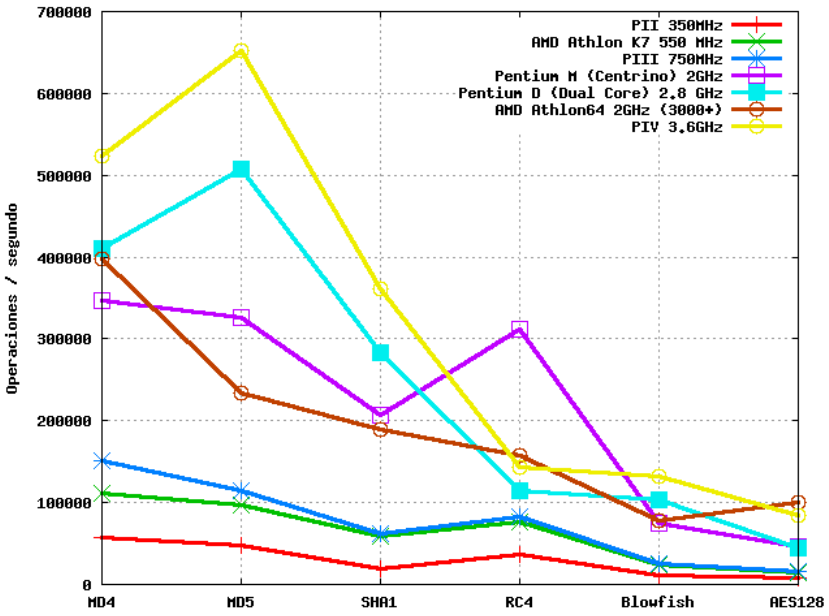

Figura 4.3: Cantidad de operaciones de cifrado por segundo que llevan a cabo los equipos evaluados

M (Centrino) al cifrar con Blowfish).

Esta alta variabilidad se debe a los cambios en la arquitectura de los procesadores introducidos de una familia a otra, y al diseño de base de cada uno de los fabricantes probados, y que acaban repercutiendo en la capacidad para realizar determinadas operaciones de bajo nivel de forma más eficiente, afectando a otras. Dado que los algoritmos criptográficos se definen principalmente como conjuntos de operaciones de bajo nivel (tales como desplazamientos a nivel de bit, permutaciones de bits, etc. . . un cambio en el rendimiento de cualquiera de esas operaciones afectará enormemente al rendimiento que el procesador ofrece con dicho algoritmo criptográfico.

En otros casos, algunas optimizaciones, tales como la integración del procesador de la tarjeta wireless 802.11 con el procesador del sistema en los equipos Pentium-M (Centrino) hace que el rendimiento de los algoritmos criptográficos utilizados en esas redes (en concreto, RC4) se vea aumentado debido a que el procesador del sistema cuenta con un procesador dedicado y optimizado para dichos algoritmos.

En cuanto a los algoritmos de firma digital y verificación, en la Tabla 4.6 y en la Figura 4.4 podemos observar cuáles son los resultados obtenidos. Como podemos observar, el equipo con un procesador AMD Athlon64 ofrece un rendimiento muy superior al resto, algo que sorprende a tenor de los resultados obtenidos con los algoritmos de cifrado y resumen criptográfico.

Tabla 4.6: Comparación del rendimiento de varios equipos al cifrar datos, en miles de operaciones por segundo

	1	2	3	4	5	6	7
RSA512 Firma	194	414	517	1.414	1.165	3.811	1.405
RSA512 Verif.	2.007	5.053	5.699	16.102	13.317	46.255	16.478
DSA512 Firma	230	544	631	1.794	1.429	6.472	1764
DSA512 Verif.	140	458	528	1.549	1.207	5.719	1514

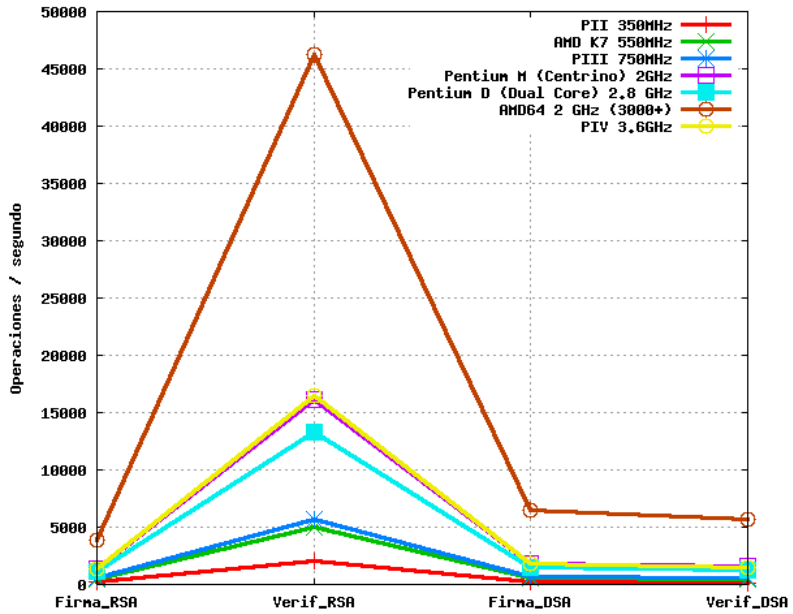

Figura 4.4: Cantidad de operaciones de firma y verificación que llevan a cabo los equipos evaluados

Asimismo, también es destacable el hecho de que el Pentium M y el Pentium IV ofrecen un rendimiento prácticamente idéntico, pese a la diferencia de velocidad y a lo dispares que son los resultados obtenidos en las pruebas anteriores.

4.2.2. Resultados del análisis

Como se puede deducir del apartado anterior, la utilización de mecanismos de seguridad hace que los protocolos de seguridad requieran de mayores recursos al establecer conexiones y proteger la información que por ellas se transmita. Este hecho puede ser utilizado por atacantes para lanzar ataques de denegación de servicio contra nuestra infraestructura de red de múltiples formas diferentes. Algunos ejemplos podrían ser:

- **Solicitar el establecimiento de nuevos túneles criptográficos y abortar la operación cuando todavía no se ha completado la negociación**, con el consiguiente gasto de ciclos de CPU y memoria.

- **Envío de falsos paquetes de los protocolos de negociación informando de falsos errores** forzando renegociaciones de las claves en el mejor de los casos, y consiguiendo cerrar el túnel criptográfico en el peor.

Además, algunos ataques podrían ser lanzados por usuarios legítimos de la arquitectura, incluso inconscientemente. Por ejemplo, como ya se ha comentado, en la versión 2 de IKE las partes no negocian los valores de caducidad de las claves, por lo que una implementación configurada con valores extremadamente bajos puede forzar a ambas partes a utilizar más recursos en renegociar información criptográfica que en proteger información.

Aunque algunos de estos ataques de denegación de servicio pueden no llegar a evitar que la implementación ofrezca servicio a los usuarios legítimos, sí que pueden lograr que la calidad del servicio ofertado sea de muy baja calidad (reducción del ancho de banda, retardos, etc. . .), consiguiendo que los usuarios eviten utilizar la arquitectura de seguridad para poder llevar a cabo sus tareas, como ya ha ocurrido con otras herramientas anteriormente ([6], [150]).

Con todo lo visto en esta sección podemos elaborar una lista de los aspectos que es interesante evaluar en una implementación de un protocolo o arquitectura de seguridad en relación con el rendimiento, relacionando cada elemento con aquellas características del sistema con las que está relacionado, y otros elementos que pueden influir en dichos aspectos.

4.2.3. Ancho de banda

El ancho de banda que puede ofrecer una implementación de un protocolo de seguridad es una medida fundamental del rendimiento que puede ofrecer dicha implementación. Como podemos deducir, la autenticación y negociación de nuevas claves no afectan realmente al ancho de banda que puede ofrecer una implementación, ya que son operaciones que tienen lugar en momentos muy concretos de todo el proceso de intercambio de información. Por lo tanto, serán la implementación que se haya realizado de los protocolos de protección de la información la que debamos evaluar.

Dado que estos protocolos son aplicaciones criptográficas de los algoritmos y técnicas negociados en fases anteriores, nos encontramos con que la mayor limitación del sistema en cuanto al ancho de banda será (además de aspectos físicos de la propia red, como el tipo de red utilizado, el cableado, las tarjetas de red, etc....) la potencia de la(s) CPU(s) del sistema. Como vimos anteriormente, las operaciones criptográficas tienen mayores requisitos de tiempo de CPU que su equivalente "en claro", por lo que el principal recurso que afectará al ancho de banda será la CPU. Para poder evaluar cuál es el máximo ancho de banda que una implementación puede ofrecer, deberemos utilizar un único túnel criptográfico por el que se transmita información tan rápido como sea posible (en el apartado 4.4 se elaborará con más detalle las condiciones y parámetros que deben regular dicha transmisión de información).

Adicionalmente, también hemos visto que las necesidades de los protocolos de seguridad en general no son únicamente tiempo de CPU, sino también memoria para poder llevar a cabo las operaciones criptográficas. Sin embargo, las necesidades globales de un único túnel criptográfico no representan grandes problemas en proporción con el cifrado de información, incluso para los dispositivos con mayores limitaciones. Por este motivo será necesario evaluar cómo afecta la existencia de múltiples túneles criptográficos simultáneos al ancho de banda que puede ofrecer una implementación concreta. Por cada túnel criptográfico deberá transmitirse el volumen de tráfico equivalente al reparto equitativo del ancho de banda de un único túnel entre todos los túneles que se establezcan. De esta forma estaremos en condiciones de evaluar el impacto de la memoria del sistema analizado en el rendimiento.

Evaluar el rendimiento de cada una de las suites criptográficas soportadas por un dispositivo nos ayudará a seleccionar aquellas suites que, ofreciendo el nivel de seguridad deseado, son capaces de ofrecer un mayor ancho de banda a cada usuario, lo que, además de las propias ventajas de esta configuración, hace que el sistema presente un mejor comportamiento ante ataques de denegación de servicio contra la implementación.

Por último, debemos destacar que el rendimiento de diferentes suites

criptográficas es completamente diferente en diferentes plataformas hardware, diferentes sistemas operativos, etc. . . ., como hemos podido observar a la luz de los resultados obtenidos en el Apartado 4.2.1.1. Además, no es posible realizar suposiciones, por muy generales que sean, referente al rendimiento que determinados algoritmos pueden ofrecer, ya que debido a optimizaciones o implementaciones defectuosas podemos encontrarnos con que algoritmos que debían ofrecer sobre el papel un rendimiento mejor que otros resultan ser mucho más lentos. Por este motivo, el ancho de banda que una implementación puede ofrecer deberá ser evaluado para todos las suites criptográficas (algoritmo de cifrado, función resumen, método de autenticación) en las que se esté interesado conocer el rendimiento.

4.2.4. Máximo número de túneles criptográficos establecidos

Dado que en el establecimiento de cada túnel criptográfico se negocian unos determinados parámetros que posteriormente serán utilizados para proteger la información, la cantidad de túneles que una implementación determinada puede mantener simultáneamente es también limitada, principalmente por la memoria de la que pueda disponer para almacenar dichos parámetros negociados. El uso de la CPU en este proceso no es tan importante, ya que nuestro principal interés en este momento es la utilización de la memoria como límite a conexiones futuras. De esta forma estaremos en condiciones de evaluar la cantidad de túneles que un dispositivo es capaz de aceptar sin descartar ninguna conexión.

Para evaluar este importante aspecto será necesario proceder al establecimiento de nuevos túneles criptográficos hasta encontrar el límite en el que la implementación estudiada debe descartar alguna conexión (bien la nueva conexión entrante, bien una conexión establecida anteriormente). Es importante resaltar que por los túneles que se establezcan no deberá transmitirse información, ya que dicha transmisión requerirá de memoria que no será empleada en aceptar nuevas conexiones, por lo que el resultado que obtendremos no será fiable. Por este mismo motivo es necesario que los establecimientos de conexión estén espaciados una cantidad prudencial de tiempo, para así evitar que las operaciones criptográficas asociadas a la negociación de un túnel afecten al establecimiento de otros túneles, especialmente cuando la cantidad de túneles establecidos es elevada.

En cuanto a la influencia de la suite criptográfica que se utilice, al igual que ocurría en el caso del ancho de banda, es necesario realizar diferentes pruebas y análisis utilizando todas las suites en las que estemos interesados. Sin embargo, al contrario de lo que ocurría en el ancho de banda, el impacto de elegir una suite u otra en los resultados no debería ser (a priori) tan acusado, ya que la mayoría de la información que se utiliza en las fases de negociación de los protocolos de seguridad tiene un tamaño fijo, indepen-

dientemente de la suite criptográfica, protocolos que se protejan, etc....

4.2.5. Capacidad de establecimiento de nuevos túneles criptográficos

Como complemento al máximo número de túneles criptográficos que nuestra implementación es capaz de establecer, debemos conocer cuáles son las capacidades de establecimiento de nuevos canales seguros, es decir, cómo se comporta nuestra implementación cuando debe hacer frente al establecimiento simultáneo de múltiples túneles criptográficos. Dado que en este aspecto están involucrados tanto CPU como memoria, podemos ver cómo se aunan parte de los análisis realizados para el ancho de banda con el número máximo de túneles.

Por lo tanto, será necesario establecer ráfagas de establecimientos de conexión y evaluar cuál es el tiempo que la implementación del protocolo de seguridad necesita para establecer los túneles criptográficos y estar disponible para la siguiente ráfaga, hasta completar el número máximo de túneles que se definió anteriormente. En el caso de que alguna de las conexiones se rechace, deberemos iniciar las pruebas desde el principio, ya que el sistema no es capaz de procesar las conexiones al ritmo que se le está exigiendo.

Este aspecto es importante de cara a conocer cuáles son las capacidades de funcionamiento de nuestro sistema, de cara a ayudarnos a realizar una planificación de la infraestructura que sea fiable y que nos evite el llevar a cabo inconscientemente un ataque de auto denegación de servicio en nuestra propia red. Por ejemplo, un escenario en el que múltiples clientes deben conectarse de forma segura a una red corporativa puede sufrir de un ataque como el comentado si alguno de los usuarios, conscientemente o inconscientemente, decide crear un túnel criptográfico nuevo por cada conexión nueva con la red corporativa, en lugar de proteger todas esas conexiones con el mismo túnel.

Por lo tanto, la medición de este parámetro de rendimiento deberá hacerse en condiciones cercanas a la realidad, en las que por los túneles previamente establecidos hay tráfico que está siendo protegido, y que requiere de ciertos recursos de la implementación de IPsec para poder continuar con su flujo de intercambio de información. Consiguientemente, un factor importante en estas pruebas es el tráfico que se transmite por cada uno de los túneles criptográficos establecidos, ya que en este caso no estamos evaluando únicamente la capacidad de tener los túneles establecidos, sino la capacidad de responder a ráfagas de trabajo, lo que implica tiempo de CPU (para llevar a cabo las operaciones criptográficas asociadas a la negociación), y que por lo tanto se verá afectada por la ocupación de la CPU al cifrar y descifrar el tráfico.

4.2.6. Tiempo de proceso

El último de los aspectos que es necesario evaluar para completar el estudio del rendimiento es el del tiempo necesario para establecer un túnel criptográfico, o dicho de otra manera, cuál es la sobrecarga que introducen la(s) fases de negociación, así como las operaciones de cifrado y descifrado en el protocolo que se utilice. De esta forma se evaluará cuál es el impacto de proteger con el protocolo elegido aquellos servicios que son dependientes del tiempo, tales como la transmisión de contenidos multimedia (videoconferencia, voz sobre IP, etc. . . .) o aplicaciones de tiempo real. Con esta información nos encontraremos en condiciones de decidir si la sobrecarga que se introduce es aceptable para los servicios que se desean proteger, o si por el contrario dicha sobrecarga haría el servicio inservible.

Dado que algunos protocolos, como IKE en IPsec, permiten la reutilización de parámetros negociados para posteriores negociaciones (en concreto IKE permite reutilizar material de la Fase 1 para negociaciones en la Fase 2, y a su vez también es posible reutilizar la negociación de la Fase 2 para múltiples túneles ESP o AH), es necesario conocer cuál es la sobrecarga que introduce cada fase por separado, así como el conjunto completo de negociaciones. De esta forma nos encontramos en condiciones de decidir cuál de las configuraciones de agrupamiento y reutilización que ofrece IPsec es la que más se ajusta a nuestras necesidades.

Dado que en este caso el único factor involucrado es la capacidad para realizar las operaciones criptográficas de las diferentes negociaciones de la forma más rápida posible, la potencia de cálculo de la implementación es el único aspecto que se evalúa. Por este motivo, y como ya se ha analizado anteriormente, es necesario realizar múltiples pruebas en las que se evalúen los tiempos necesarios cuando se utilizan diferentes suites criptográficas.

4.3. Medición de los factores de rendimiento

4.3.1. Ancho de banda

Para evaluar el ancho de banda que la implementación de un protocolo o arquitectura de seguridad puede ofrecer es necesario conocer qué información acerca del ancho de banda nos interesa, ya que el máximo ancho de banda depende de factores tales como el tipo de tráfico que se transmite, si el tráfico es entrante o saliente (ya que las operaciones de descifrado son más complejas computacionalmente y por lo tanto requieren de mayor cantidad de recursos), etc. . .

Por lo tanto, una primera necesidad para llevar a cabo esta medida es la de conocer qué tipo de tráfico queremos utilizar para llevar a cabo la evaluación del rendimiento. Mientras que tráfico “genérico” (como por

ejemplo sólo tráfico UDP, o sólo tráfico TCP) nos dará información acerca de las capacidades máximas de nuestro sistema, dichos modelos de tráfico son irreales, y por lo tanto no son útiles desde un punto de vista práctico. Por otro lado, un modelo de tráfico más realista (por ejemplo, 20 % de tráfico UDP y 80 % de tráfico TCP), aunque más realistas, presentan el problema de que su medición nunca ofrecerán información acerca de las máximas posibilidades de la implementación, sino que únicamente presentan datos concretos del modelo utilizado.

Para solventar este problema, en la metodología se apuntarán ciertos modelos o perfiles de tráfico que deberán ser utilizados para las pruebas de rendimiento de forma obligatoria para obtener información acerca de las máximas capacidades del sistema, y se permitirá al usuario utilizar sus propios modelos de tráfico para que así obtenga información concreta acerca del ancho de banda que puede obtener de acuerdo a sus patrones de utilización de la red de comunicaciones actuales. Con fines meramente orientativos, se incluirán en la metodología algunos patrones de tráfico que representan aproximadamente perfiles comunes de utilización de los recursos de red. Un análisis más detallado de los perfiles de tráfico se realizará en el apartado 4.4.

El siguiente problema que se nos plantea es cómo llevar a cabo la medida de estos valores. Dado que es posible utilizar protocolos como UDP, que no cuentan con medidas de notificación de pérdida de paquetes o de congestión de la red, los únicos puntos en los que se puede realizar la medida es en los puntos finales de la comunicación, ya que únicamente el tráfico que haya llegado a su destino ha sido procesado por la implementación evaluada. En la Figura 4.3.1 podemos ver un ejemplo ilustrativo: El ancho de banda del tráfico A se computaría en el equipo 2, mientras que el ancho de banda del tráfico B será mediría en el equipo 1. El ancho de banda total será la suma de ambos valores.

Para evitar sobrecargar innecesariamente los equipos que establecen los túneles criptográficos únicamente se establecerá un túnel criptográfico con el que se protegerá todo el tráfico que se genere. Adicionalmente, no es recomendable utilizar más de un equipo origen y destino en cada extremo de los túneles, con el fin de evitar sobrecargas en la red que pudieran alterar los resultados de nuestras mediciones.

4.3.2. Máximo número de canales seguros simultáneas

Al evaluar el máximo número de conexiones seguras que una implementación de un protocolo de seguridad puede mantener abiertas simultáneamente, es necesario evaluar cuáles son los parámetros que influyen en la capacidad de negociar canales seguros. Como en este caso no estamos evaluando la rapidez de negociación de nuevos túneles criptográficos, el factor

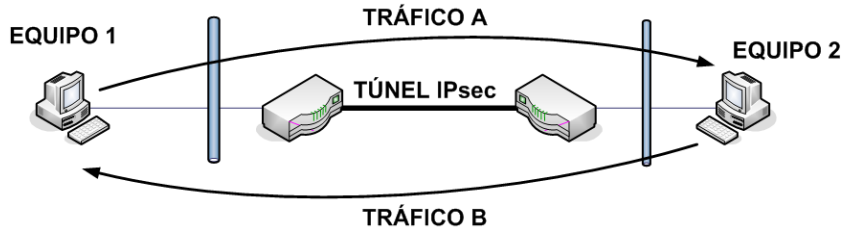

Figura 4.5: Esquema de medición del ancho de banda.

que más influencia tiene es la memoria del dispositivo. Sin embargo, al llevar a cabo un análisis en profundidad nos encontramos con que, al acercarse a los límites de canales de seguridad que pueden establecerse, la respuesta de las implementaciones va haciéndose cada vez más lenta. Por este motivo es necesario que la evaluación de la cantidad de túneles criptográficos que puede establecer la implementación que se está estudiando se lleve a cabo utilizando la menor cantidad de recursos posibles para cada asociación de seguridad.

La solución a este problema pasa por establecer canales seguros en los que, una vez establecido el túnel criptográfico, no se transmite información por dicho túnel. Sin embargo, la comunicación sigue establecida (mediante la utilización de un protocolo orientado a conexión, como TCP) y por este motivo no es viable que ninguna de las implementaciones fuerce el cierre de ninguna de las conexiones seguras establecidas. Adicionalmente, para evitar que se produzcan renegociaciones de las claves criptográficas mientras se lleva a cabo esta prueba, se configurarán ambas implementaciones de forma que la caducidad de las claves sea un valor muy alto en el tiempo, superior al que se empleará en este análisis.

Un problema adicional que nos podemos encontrar es que, a partir de cierto volumen de conexiones seguras establecidas, el establecimiento de nuevas conexiones requiere de un tiempo lo suficientemente elevado como para que el solapamiento de dos solicitudes de establecimiento de conexión conlleve el rechazo de una de ellas, aun cuando la implementación sea capaz de establecer más túneles criptográficos.

Para solventar este problema en la metodología se propone la repetición de esta prueba separando las solicitudes de establecimiento de nuevas conexiones seguras con tiempos crecientes. En el momento en que varias pruebas generan los mismos resultados, habiendo espaciado sus solicitudes unos intervalos de tiempo mayores cada vez, podemos concluir que la cantidad alcanzada de conexiones seguras es el máximo número que puede soportar la implementación.

Hay que destacar que, dado que la cantidad de conexiones seguras que puede soportar una implementación de seguridad puede ser muy elevado, la

forma de evaluar este parámetro requerirá el establecimiento de conexiones seguras independientes para cada comunicación que se establezca entre las redes protegidas por las implementaciones, sin que se agrupe el tráfico de acuerdo a ningún parámetro, con el fin de facilitar la prueba y evaluación de este parámetro.

4.3.3. Capacidad de establecimiento de canales seguros

Como combinación de las medidas de ancho de banda y máximo número de canales seguros surge la medición de la capacidad de establecimiento de conexiones seguras. Esta medida combina parte de las características de la medición del ancho de banda (en cuanto a que será necesario generar tráfico para controlar el ancho de banda en cada túnel criptográfico), y parte de las características del establecimiento de conexiones seguras.

Sin embargo, la problemática para medir este factor es menor que en casos anteriores:

- En este caso el tipo de tráfico que se utiliza no es importante, ya que al dividir el ancho de banda entre múltiples conexiones ningún canal llegará a tener por sí mismo un porcentaje de uso de la red tan elevado que se vea afectado por el tipo de protocolo utilizado[5]. Sí que debe utilizarse un protocolo orientado a conexión para facilitar la medición del tráfico cursado.

- El tiempo que se debe dejar entre las solicitudes de negociación de nuevas conexiones es uno de los factores que se pretenden medir en esta prueba, por lo que no es necesario llevar a cabo ninguna suposición o introducir un valor preliminar a la espera de los resultados.

Por estos motivos, el análisis de la capacidad de establecimiento de conexiones seguras se evaluará estableciendo una comunicación entre las redes protegidas que obligue a solicitar una nueva conexión para proteger esa comunicación, y procediendo a transmitir por dicho túnel criptográfico información, de tal forma que se utilice un ancho de banda predefinido.

4.3.4. Tiempo de proceso

Al evaluar el tiempo de proceso de cada uno de los protocolos que conforman un protocolo o arquitectura de seguridad nos encontramos con el problema de que la arquitectura de red utilizada para conectar los dispositivos que establecen los túneles criptográficos introduce retardos en las

[5]Habitualmente todos los protocolos de red pueden utilizar un máximo de, al menos, el 60 % del ancho de banda máximo de la red sin tener que recurrir a optimizaciones de ningún tipo

comunicaciones. Este factor nos impedirá evaluar cuanto tarda exactamente la implementación en llevar a cabo todas las operaciones necesarias para establecer el túnel criptográfico, ya que no podemos tomar medidas fiables en nuestra implementación, ni podemos acceder a los registros de la implementación analizada. Sin embargo, podemos obtener información acerca de cuánto tiempo tarda, visto desde la implementación que controlamos, lo que viene a representar el tiempo tal y como lo percibe el usuario, no el procesador de red.

Por lo tanto, aunque utilizaremos nuestra implementación para ofrecer información acerca del tiempo empleado para llevar a cabo la negociación de cada una de las fases de negociación y para comenzar la transmisión de información protegida mediante los parámetros negociados, los valores que ofrecen información más interesante son los que provengan de capturar el tiempo mínimo desde que se envía un mensaje que requiere del establecimiento de una nueva asociación de seguridad hasta que se recibe la respuesta a dicho mensaje.

Este factor de rendimiento es muy dependiente de la infraestructura hardware que se esté utilizando para llevar a cabo las pruebas: una infraestructura mínima, con conexiones directas entre los equipos que establecen el túnel criptográfico implicará un valor más realista en cuanto a tiempo de proceso empleado por la implementación, mientras que una infraestructura más compleja, pero más parecida a la que utilicen los usuarios del sistema ofrecerá información real acerca de los retardos que experimentarán los usuarios. Esta información puede ser muy importante en el caso en que se desee utilizar el protocolo o arquitectura de seguridad para proteger tráfico de red con necesidades de tiempo real, como Voz sobre IP.

Como ya se ha comentado, la evaluación de este factor vendrá dada por un lado por los tiempos que registre la implementación bajo nuestro control, y por otro, por los tiempos registrados por una aplicación de usuario desde que envía un mensaje a otro equipo al otro lado del túnel criptográfico hasta que recibe la respuesta.

4.4. Aspectos a tener en cuenta

Al igual que ocurría con la evaluación de la conformidad de la implementación del protocolo o arquitectura de seguridad, al medir las capacidades y el rendimiento de dicha implementación entran en juego factores que pueden influir en los resultados, por lo que es necesario conocer y estudiar cómo hacer frente a dichos factores. Estos factores son los perfiles de tráfico que se utilizan al medir el ancho de banda, y la arquitectura de red que se utiliza para llevar a cabo la evaluación.

Al estudiar el rendimiento que puede ofrecer un dispositivo de red (es-

pecialmente los que desarrollan protocolos de seguridad) es importante remarcar la importancia de los perfiles de transmisión de información que se utilizan durante las evaluaciones anteriores, y, aunque no es un aspecto a evaluar en sí mismo, su impacto en otros factores es lo suficientemente relevante como para estudiarlo. Por perfiles de tráfico estamos haciendo referencia a un modelo de utilización de las redes de comunicaciones, en el que se especifican los protocolos de comunicaciones que se están utilizando para transmitir la información, así como a la relación entre la cantidad de datos enviada por cada entidad de la comunicación. Así, por ejemplo, podríamos hablar del perfil *UDP simétrico* refiriéndonos a un perfil que utiliza UDP para la transmisión de datos, y en el que ambas partes están enviando la misma cantidad de información.

Un aspecto de los perfiles de tráfico que afecta al rendimiento que ofrecen los dispositivos de red (en este caso, las implementaciones del protocolo de seguridad) es la cantidad de datos que se envía en cada paquete TCP, UDP, ... Si estamos interesados en conocer cuál es el máximo rendimiento de la implementación deberemos utilizar paquetes en los que el tamaño del campo de datos es muy cercano al MTU para la tecnología de red que se está utilizando. Por contra, paquetes en los que la carga útil es muy reducida nos dan información acerca de la sobrecarga que introduce IPsec en la red. Por lo tanto, al definir un perfil es necesario indicar también qué tamaño de bloques de datos se están utilizando para enviar los mensajes a través de la red.

Esta decisión contrasta con la propuesta de la metodología para el estudio del rendimiento de dispositivos de red del IETF ([19]), en la que las pruebas de medición del ancho de banda se realizan con múltiples tamaños de paquete. Sin embargo, al encapsular la información utilizando algoritmos de cifrado cuyos bloques de salida son de tamaño constante y al incluirse la posibilidad de rellenar campos de datos con valores nulos para aumentar la entropía, nos encontramos con que, independientemente del tamaño del paquete origen, los paquetes de salida tienen un tamaño fijo que escapa a nuestro control. Por este motivo nuestra propuesta no requiere de la realización de las pruebas con múltiples tamaños de bloque, si no que el tamaño de bloque viene determinado por el perfil de tráfico que se utilice.

A pesar de que en la metodología se definen algunos perfiles para obtener información que sea relevante, y otros que representan usos genéricos de las redes de comunicaciones, siempre será recomendable el realizar pruebas adicionales con perfiles de tráfico tan parecidos al tráfico real que será protegido por IPsec como sea posible. Ejemplos de algunas herramientas que pueden servir para definir estos perfiles de tráfico a partir de tráfico real son [152] y [47].

En cuanto a la arquitectura de red que se utilice, es indudable que las capacidades de los dispositivos de red utilizados para interconectar los dispo-

sitivos que implementan el protocolo o arquitectura de seguridad, la cantidad de dispositivos intermedios, la distancia entre los equipos y los dispositivos de red, etc... pueden influenciar algunos de los parámetros que se deben medir. Por este motivo, es esencial comprobar que la infraestructura de red que se utilizará puede soportar la cantidad de tráfico que se enviará por ella, así como comprobar que los equipos que se utilizarán para enviar y recibir tráfico pueden soportar la cantidad de canales seguros necesaria para conocer el máximo que soporta la implementación que se estudia.

El problema que se nos presenta es que, dado que aún no se dispone de información fiable acerca del rendimiento de la implementación, no conocemos a priori dónde están los límites de la implementación que se estudia. Para solventar este problema se puede optar por dos enfoques diferentes: *Utilizar la información de rendimiento suministrada por el fabricante* como valor orientativo de los máximos valores que podremos obtener, o bien *calcular el rendimiento máximo teórico* que se puede ofrecer teniendo en cuenta la cantidad y el tipo de interfaces de red de los que dispone el sistema.

Cada uno de estos métodos tiene sus limitaciones, ya que, por ejemplo, al utilizar los datos proporcionados por el fabricante podemos encontrarnos que las únicas especificaciones disponibles están basadas en el procesador criptográfico, pero no en la arquitectura final de la implementación[6]. Si se decide utilizar el segundo método hay que prestar atención a las limitaciones de configuración en algunas plataformas, especialmente hardware, que limitan la cantidad de interfaces físicos cuyo tráfico puede ser protegido, siendo de este modo que únicamente estos interfaces son los que han de tenerse en cuenta para el cómputo del máximo ancho de banda que podría ofrecer la implementación.

Un aspecto adicional que no se ha mencionado hasta el momento es el estado de la red en el momento en que se llevan a cabo las pruebas, ya que aspectos tales como el retardo de la red, la fragmentación, la pérdida de tramas o las colisiones que pueda haber afectarán a los valores de rendimiento que obtengamos. Por un lado, este hecho podría ser obviado ya que el uso de infraestructura de red dedicada para llevar a cabo las pruebas necesarias evitaría que se den condiciones hostiles en la red que no hayan sido generadas por las propias pruebas, por lo que no sería necesario tenerlas en cuenta. Sin embargo, dado que el objetivo de estas pruebas es conocer cuál es el comportamiento de la implementación estudiada y obtener información realista del rendimiento que se puede esperar de ella, es necesario

[6]Por ejemplo, es común encontrarse con especificaciones en las que el fabricante asegura que su implementación puede proteger un ancho de banda de hasta 400 Mbps, cuando el dispositivo o sistema al que se refieren esas especificaciones únicamente cuenta con una tarjeta de red de 100 Mbps. En estos casos es recomendable utilizar el segundo método, ya que las especificaciones del fabricante no nos aportan información útil para nuestras estimaciones

incluir estos factores en las pruebas. Con el fin de flexibilizar la utilización de estos parámetros y poderlos conjugar con otros parámetros tales como protocolos utilizados (parámetros de los que ya se ha hablado anteriormente), se propone la inclusión del estado de la red en los propios perfiles de utilización de la red.

De cara a la realización de las pruebas, la inclusión del estado de la red como factor a tener en cuenta durante la realización de las pruebas hace que debamos considerar cuáles son las opciones que se nos presentan para poder controlar estos aspectos de la red, ya que la utilización de infraestructura de red dedicada hará que no se produzcan espontáneamente, y de forma ajena al tráfico correspondiente a las pruebas, las condiciones para que la red fragmente los paquetes de datos, aumente la latencia de la red (tanto de forma simétrica como asimétrica), o los dispositivos comiencen a descartar paquetes para poder hacer frente a la carga de la red. Las soluciones que se plantean en este momento son dos:

- **Generación de tráfico en la red para que se den las condiciones necesarias**. Mediante la utilización de equipos situados en cada uno de los segmentos de red utilizados en la infraestructura de red utilizada para llevar a cabo las pruebas de rendimiento, sería posible generar tráfico entre dichos equipos para generar las condiciones de red en las que deseemos llevar a cabo las pruebas, como aparece reflejado en la Figura 4.6. El tráfico se generaría entre equipos del mismo segmento de red (señalado en rojo) cuando se desee alcanzar una situación concreta en un segmento de red determinado, o entre segmentos de red diferentes (señalado en amarillo) para forzar las condiciones en un conjunto de segmentos concretos.

  Frente a la indudable ventaja que representa el hacer la evaluación del rendimiento en las condiciones de red reales, surgen importantes inconvenientes, de los que la gran cantidad de recursos necesarios para poder forzar ciertas condiciones (como por ejemplo, el descarte de paquetes o el incremento sustancial del retardo) en redes de alta velocidad, y la parametrización del tráfico que se genera de forma manual, debiendo llevar a cabo para cada red diferente cuáles son los valores de generación de tráfico que ocasionan los niveles de retardo y pérdida de tramas que deseamos establecer en la red.

- **Utilización de dispositivos de red adicional que fuerce las condiciones de red deseadas**. Haciendo uso de dispositivos de red adicionales que hagan las funciones de puentes entre segmentos de red diferentes y que sean capaces de generar artificialmente las condiciones de red necesarias al llevar a cabo esa interconexión entre los segmentos de red, es posible forzar a que el túnel criptográfico entre los segmentos de red involucrados se desarrolle en las condiciones deseadas. En este

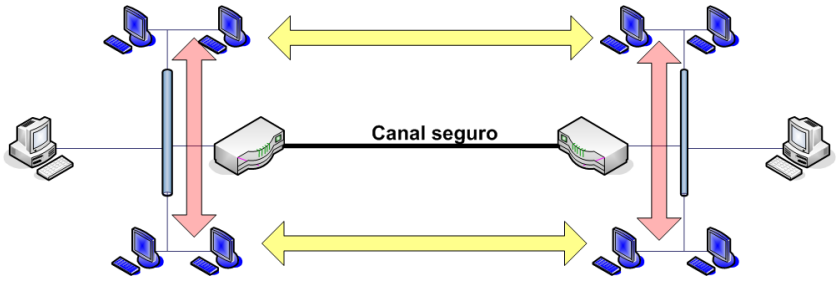

Figura 4.6: Arquitectura de red necesaria para generar tráfico y hacer que aparezcan en la red las condiciones deseadas

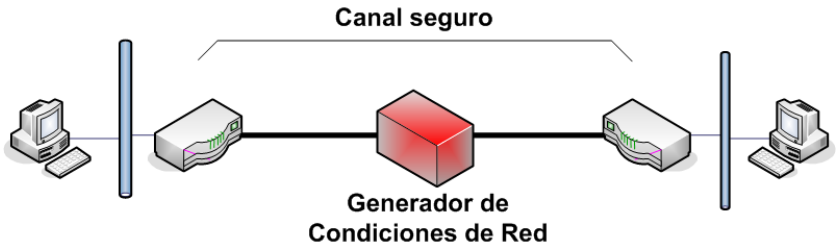

Figura 4.7: Arquitectura de red en la que un dispositivo de red adicional genera las condiciones deseadas

caso el esquema de red básico para llevar a cabo las pruebas sería el mostrado en la Figura 4.7, en el que el dispositivo en rojo es el encargado de generar artificialmente el retardo en la red, la pérdida de paquetes, la fragmentación etc...

En este caso, la principal desventaja es que las condiciones de red que se están forzando son irreales, por lo que, dependiendo de la calidad del dispositivo utilizado y de la flexibilidad a la hora de configurar dicho dispositivo, existe la posibilidad de que los resultados que se obtengan en las pruebas se vean alterados y no sean válidos para obtener información acerca del rendimiento de la implementación del protocolo o arquitectura de seguridad. Sin embargo, las ventajas que ofrece esta solución son las derivadas de los escasos requisitos hardware y software adicionales que son necesarios para desplegar esta solución, así como la posibilidad de seleccionar exactamente cuáles son las condiciones que se desean utilizar en la red, incluyendo el margen de error con el que se trabaja.

Otro problema que se nos presenta, aunque con un impacto en los resultados mucho menor, es el tiempo necesario para llevar a cabo las pruebas. Con el fin de conseguir que los resultados obtenidos sean lo suficientemente relevantes estadísticamente, se deberán tener en cuenta aspectos amplia-

mente aceptados referentes a la capacidad de repetición, la varianza y la representatividad estadística. Esto nos lleva a que, partiendo de los valores y estudios que se pueden ver en [19], [144] y [84]. Por lo tanto, cada prueba deberá repetirse al menos 3 veces, (recomendable 5) para obtener valores con una fiabilidad mucho mayor, lo que implica que la realización de los tests utilizando todas las posibles suites criptográficas soportadas por la implementación supondría un tiempo considerable. Por este motivo es necesario que se seleccionen aquellas suites criptográficas que más interesantes nos resultan. Es de destacar el hecho de que, tras haber llevado a cabo las pruebas de conformidad con el estándar y corrección criptográfica descritos en la sección 3.2, únicamente serán candidatas a ser seleccionadas aquellas suites criptográficas que hayan resultado ser criptográficamente correctas, ya que el resto de dichas suites no pueden ser evaluadas desde la implementación de referencia.

La selección de las suites criptográficas deberá realizarse indicando para cada una de ellas en qué fases del protocolo o arquitectura de seguridad deberá ser evaluada: negociación o protección de la información. Esto se debe a que es posible que no todas las suites criptográficas deban utilizarse en todas las fases de protocolo, reduciendo de esta manera la cantidad de pruebas a llevar a cabo durante la evaluación del rendimiento.

Capítulo 5

Metodología de Validación y Evaluación Remota de Protocolos y Arquitecturas de Seguridad

5.1. Introducción

Una vez estudiados y revisados aquellos factores que deben ser analizados a la hora de llevar a cabo estudios acerca de la conformidad (capítulo 3) y el rendimiento (capítulo 4) de implementaciones de protocolos y arquitecturas de seguridad, nos encontramos en situación de presentar la metodología de validación y evaluación remota de protocolos y arquitecturas de seguridad, que representa el núcleo en torno al que se estructura esta tesis. La metodología se divide de forma que inicialmente se presentan las definiciones de términos que utilizarán en la metodología (sección 5.2), y un análisis acerca de la conformidad y el rendimiento (sección 5.3). Posteriormente se presentan 8 fases en las que se abarcan desde las tareas preliminares a llevar a cabo hasta los procedimientos de finalización una vez terminada la validación y evaluación (secciones 5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 5.10 y 5.11. Estas fases se desarrollarán de forma secuencial, aunque la ejecución de cada una de ellas es opcional, si los objetivos que se persiguen no se ajustan a los del evaluador[1].

[1]Por ejemplo, la fase de documentación preliminar puede ser omitida si los conocimientos básicos acerca del protocolo o arquitectura de seguridad ya se poseen

5.2. Definiciones

Conjunto de Pruebas: Un conjunto completo de pruebas individuales que se llevan a cabo para obtener información acerca de una determinada característica o funcionalidad en la IEP.

Éxito (resultado): Resultado de las pruebas de conformidad en las que la IEP demuestre ser conforme al estándar en el aspecto evaluado.

Fallo (resultado): Resultado de las pruebas de conformidad en las que la IEP demuestre no ser conforme al estándar en el aspecto evaluado, o genere eventos no contemplados en la especificación del protocolo o arquitectura.

IDR: Implementación de referencia; aquella que es conforme al estándar.

IEP: Implementación en pruebas.

Medios de Prueba: Combinación de equipos y procedimientos que pueden llevar a cabo la validación o evaluación deseada.

Perfil de Tráfico: Conjunto de características que definen un uso de la red determinado, así como el estado de la propia red.

Prueba de Capacidad: Prueba que determina si una determinada característica se encuentra implementada en la IEP.

Prueba de Comportamiento: Prueba que determina si uno o más requisitos de conformidad dinámicos (ver sección 5.3) se cumplen en la IEP.

Pruebas de Conformidad: Validación del nivel de conformidad con el estándar de una IEP.

Pruebas de Rendimiento: Evaluación del rendimiento de la IEP en diversas condiciones de operación.

Registro de Pruebas: Registro en formato legible por las personas de los resultados obtenidos en la realización de una prueba o un conjunto de pruebas.

Repetibilidad de los Resultados: Características de un conjunto de pruebas, tal que repetidas ejecuciones del mismo conjunto de pruebas sobre la misma IEP en las mismas condiciones, conducen a los mismos resultados.

Resultado Obtenido: Resultado de una prueba de conformidad obtenido para una IEP concreta.

Resultado Previsto: Resultado de una prueba de conformidad de acuerdo con las especificaciones del protocolo o arquitectura.

Sistema de Pruebas: Conjunto de equipos, dispositivos y software utilizado para realizar las diferentes pruebas sobre la IEP.

5.3. Conformidad y Rendimiento

Una implementación de un protocolo o arquitectura de seguridad es conforme a la especificación de dicho protocolo o arquitectura si dicha implementación presenta todas las características y requisitos descritos en esa especificación. Estos requisitos de conformidad pueden ser **obligatorios**, cuando deben ser respetados en todo momento, **condicionales**, cuyo cumplimiento se reduce a los casos en los que se dan un conjunto determinado de condiciones, u **opcionales**, cuando su cumplimiento por parte de las implementaciones es totalmente voluntario. Adicionalmente, nos encontramos con que los requisitos de conformidad pueden estar expresados de forma positiva (se especifica lo que debe hacerse) o negativa (cuando se detalla lo que no debe hacerse). Por último, podemos comprobar cómo los requisitos de conformidad se pueden agrupar en requisitos de *conformidad estática* o *conformidad dinámica*[2].

La *conformidad estática* es aquella que establece límites a las combinaciones de características permitidas en las implementaciones de los protocolos o arquitecturas de seguridad, así como el conjunto de características mínimo que permite la interoperabilidad. Por su parte, la *conformidad dinámica* especifica el comportamiento observable que es admisible para las implementaciones del protocolo en cuestión. La definición de los protocolos de seguridad en los estándares es el caso más claro de conformidad dinámica: el uso y formato que se asigna a las unidades de información, transiciones entre estados, reglas de negociación, etc.... La conformidad dinámica establece límites a la funcionalidad de las implementaciones, por lo que define el máximo conjunto de características que una implementación puede tener.

En cuanto al rendimiento que ofrece una implementación de un protocolo o arquitectura de seguridad, los parámetros que se evalúan pueden ser **dependientes del tráfico de red** o **independientes del tráfico de red**. Aquellos *parámetros que dependen del tráfico* de red modifican sus resultados en función del tipo de mensajes que se utilizan para evaluarlo: protocolos, tamaño de los mensajes, sentido de los mensajes, ... También se ven afectados por el tráfico existente en la red en la que se realizan las pruebas, y por la saturación de dicha red. Esta variabilidad afecta a la repetibilidad de los resultados de rendimiento que se obtienen al llevar a cabo la evaluación.

[2]Esta nomenclatura es la misma utilizada por las normas ISO/IEC 9646 e ITU-T X.290

Los *parámetros de rendimiento que no dependen del tráfico* de red son aquellos en los que las características de los mensajes intercambiados no influyen en los resultados obtenidos. En la medición de estos parámetros el nivel de saturación de la red afecta de forma marginal a los resultados, por lo que la repetibilidad de las pruebas no resulta afectada.

5.4. Fase 1: Tareas Preliminares

En la fase inicial de las pruebas, se determinará cuáles son las implementaciones a validar y evaluar, quién llevará a cabo dichos procesos y qué recursos serán necesarios para completar esta tarea. Para poder llevar a cabo este proceso será preciso determinar cuál va a ser el objetivo de los análisis que se lleven a cabo, y obtener los recursos necesarios para desarrollar esos estudios.

5.4.1. Determinar el tipo de análisis

Existe la posibilidad de llevar a cabo tres tipos de análisis, dependiendo del tipo de información que se desee obtener:

Validación de la conformidad En este caso únicamente se llevarán a cabo pruebas de conformidad sobre la IEP. Se obtendrá información acerca de la conformidad de la implementación con el estándar y de las capacidades de la IEP. Se desarrollarán conjuntos de pruebas de capacidad y de comportamiento. Interpretando los resultados de varias implementaciones es posible deducir las posibilidades de interoperatividad entre ellas.

Evaluación del rendimiento Al llevar a cabo este análisis se someterá a la IEP a pruebas de rendimiento que proporcionen información acerca de la capacidad de protección de información de dicha IEP. Los análisis están compuestos de conjuntos de pruebas de comportamiento.

Validación de la conformidad y evaluación del rendimiento Es posible llevar a cabo los procesos de validación de la conformidad y evaluación del rendimiento de forma conjunta. El análisis final constará de la ejecución de cada uno de los análisis de forma secuencial.

Adicionalmente, se deberán determinar cuál es la versión del protocolo o arquitectura de seguridad que se va a evaluar, el sistema de pruebas en el que el proceso de validación y evaluación se llevará a cabo, la versión de la IDR que se utilizará y otros recursos software o de infraestructura que sean necesarios.

5.4.2. Identificación de los recursos necesarios

Es posible que no todos los recursos necesarios determinados en el apartado anterior sean accesibles al público en general, por lo que en esta fase el responsable del estudio deberá llevar a cabo las gestiones necesarias para averiguar qué herramientas y utilidades pueden dar el servicio necesario (especialmente la IDR), identificando aquellas cuya disponibilidad sea mayor. Todos los recursos que se emplearán para llevar a cabo los análisis se incluirán en un registro para su posterior comprobación en la fase final de esta metodología.

5.5. Fase 2: Documentación Preliminar

Dado que los evaluadores no deben disponer de conocimientos previos acerca del protocolo o arquitectura de seguridad cuya implementación se evaluará, es necesario que, como paso previo al diseño de las pruebas de conformidad y de rendimiento, se adquiera este conocimiento. Del mismo modo, en esta fase se incluirá la documentación acerca de los problemas de interoperabilidad más frecuentes en las implementaciones de este protocolo o arquitectura.

Se procederá también a instalar la IDR en un entorno de pruebas para adquirir conocimientos adicionales acerca de las implementaciones del estándar que amplíen los conocimientos del evaluador en este área.

Para finalizar la presente fase se documentarán los aspectos generales del protocolo o arquitectura de seguridad, así como los problemas de interoperatividad y rendimiento más frecuentes en la actualidad.

5.6. Fase 3: Análisis del Estándar

En esta fase se llevará a cabo un estudio exhaustivo del estándar que permita ampliar los conocimientos adquiridos en la fase anterior. Al finalizar este proceso, los evaluadores deben ser capaces de:

- Identificar aquellos aspectos fundamentales del estándar que deben ser evaluados tanto en lo referente a conformidad como a rendimiento.
- Identificar aquellos aspectos del estándar que deben ser evaluados debido a su importancia para la seguridad.
- Identificar aquellos aspectos del estándar que deben ser evaluados por su influencia en problemas de interoperabilidad existentes.

- Identificar aquellos aspectos del estándar que deben ser evaluados por su importancia en determinadas situaciones o arquitecturas determinadas.

- Identificar aquellos aspectos del estándar que deben tenerse en cuenta en la evaluación del rendimiento

La presente fase finalizará documentando la información obtenida en cuanto a los aspectos del estándar que deben ser incluidos en los análisis.

5.7. Fase 4: Validación de la Conformidad

Para llegar a definir un conjunto de pruebas que permitan el análisis de la IEP, es necesario identificar aquellas capacidades que aparecen recogidas en el estándar y que deben ser desarrolladas por las implementaciones de una forma determinada.

5.7.1. Identificación de los mecanismos criptográficas

Dado que muchas de las diferentes configuraciones surgirán a partir de variaciones en los mecanismos criptográficos que se utilizan, es necesario identificar cuáles son las suites criptográficas permitidas por el estándar en cada una de las fases del desarrollo del protocolo de seguridad. Adicionalmente, se identificarán aquellos mecanismos que deben ser implementados obligatoriamente y aquellos que son opcionales.

Teniendo en cuenta la constante evolución de los estándares de los protocolos, se prestará especial atención a la presencia de mecanismos opcionales en la actualidad pero que pasarán a ser obligatorios en un corto periodo de tiempo (normalmente identificados como `SHOULD+` en los estándares) y aquellos que son obligatorios pero dejarán de serlo (señalados como `MUST-`).

5.7.2. Identificación de los características obligatorias

Las especificaciones de los protocolos y arquitecturas de seguridad incluyen un conjunto mínimo de funcionalidades que deben ser incluidas en cualquier implementación de dicho protocolo o arquitectura. Este conjunto mínimo de funcionalidades debe ser identificado para poder incluir su validación en el conjunto de pruebas. Deberá prestarse especial atención a aquellas funcionalidades o mecanismos que se encuentran en evolución, bien para pasar a ser obligatorios, bien porque pronto dejarán de serlo.

5.7.3. Identificación de los características opcionales que deben ser evaluadas

En los protocolos y arquitecturas de seguridad se incluyen múltiples mecanismos y opciones que colaboran a aumentar la seguridad del sistema o a hacer el protocolo o arquitectura utilizable en diferentes arquitecturas y topologías de red. Aquellos mecanismos que, pese a ser opcionales, sean importantes para la seguridad de la información protegida, o sean necesarios para poder utilizar el protocolo en determinadas arquitecturas, deberán ser incluidos en los aspectos a analizar durante el estudio. También se incluirán en el conjunto de pruebas aquellas funcionalidades que en la actualidad presentan mayor cantidad de problemas de interoperatividad. Deberá tenerse en cuenta a la hora de determinar los resultados de conformidad que estas capacidades de las que estamos hablando son opcionales, por lo que una implementación puede ser conforme al estándar sin desarrollar ninguna de estas capacidades.

5.7.4. Diseño de pruebas

Una vez se disponga de información suficiente acerca de los aspectos del estándar que se van a validar, se procederá a definir el conjunto de pruebas que servirá para estudiar la IEP. Las pruebas de conformidad serán pruebas de capacidad y de comportamiento, según el aspecto del estándar que pretende validarse.

La definición de cada prueba constará de:

- **Objetivos**, capacidad que se pretende validar con cada una de las pruebas.
- **Obligatoriedad**, de la implementación de dicha capacidad.
- **Medios de prueba**, configuración del sistema de pruebas, y arquitectura necesaria.
- **Resultado previsto**, para las implementaciones conformes. Descripción de *éxito* y *fallo* para cada prueba concreta. Reacción ante los resultados previstos más habituales.
- **Configuraciones válidas**, variaciones de parámetros que es necesario realizar a la hora de llevar a cabo el análisis.
- **Medición de los resultados**, incluyendo los registros de pruebas.

5.8. Fase 5: Evaluación del Rendimiento

Para llegar a definir un conjunto de pruebas que permita el análisis del rendimiento ofrecido por la IEP, es preciso identificar aquellos aspectos del rendimiento del protocolo o arquitectura de seguridad que es necesario evaluar.

5.8.1. Rendimiento de los mecanismos criptográficas

El rendimiento de los diferentes mecanismos criptográficos es muy variable entre distintos sistemas, incluso de la misma arquitectura. Por este motivo es necesario identificar los diferentes conjuntos de mecanismos involucrados en cada uno de los aspectos del rendimiento que es necesario evaluar, para así poder llevar a cabo la evaluación del rendimiento de dicho aspecto considerando la utilización de cada uno de esos mecanismos.

Teniendo en cuenta la constante evolución de los mecanismos criptográficos utilizados en los protocolos de seguridad, se prestará especial atención a la presencia de mecanismos identificados como `SHOULD+` (es decir, opcionales actualmente pero obligatorios a corto plazo) y aquellos señalados como `MUST-` (obligatorios en la actualidad, pero que pronto dejarán de serlo).

5.8.2. Identificación de parámetros dependientes del tráfico

Aquellos parámetros de rendimiento que sean dependientes del tráfico que circule por la red, o de las características concretas del tráfico que se utiliza para llevar a cabo la evaluación, deberán ser identificados. En los conjuntos de pruebas que evalúen estos parámetros deberá determinarse cuáles son las condiciones en las que dicha prueba debe llevarse a cabo, con el fin de facilitar la repetibilidad de la evaluación.

5.8.3. Identificación de parámetros independientes del tráfico

Se identificarán los aspectos del rendimiento a los que el estado de la red y las característica de tráfico de prueba no afectan directamente. Los conjuntos de pruebas que evalúen estos parámetros no deben considerar el estado de la red ni el tipo de tráfico utilizado. Sin embargo, dado que es posible que al llegar a los límites de rendimiento que ofrece la IEP se produzcan variaciones por estos motivos, los conjuntos de pruebas deberán incluir los mecanismos necesarios para detectar estas situaciones, y obtener resultados repetibles.

5.8.4. Diseño de pruebas

Una vez se disponga de información suficiente acerca de los aspectos del rendimiento que se van a evaluar, se procederá a definir el conjunto de pruebas. Las pruebas de rendimiento son pruebas de comportamiento únicamente, por lo que ninguna prueba deberá considerar a la IEP como algo más que una caja blanca.

La definición de cada prueba de rendimiento constará de:

- **Objetivos**, aspecto que se pretende evaluar con cada una de las pruebas.
- **Medios de prueba**, configuración del sistema de pruebas, y arquitectura necesaria.
- **Configuraciones válidas**, variaciones de parámetros que es necesario realizar a la hora de llevar a cabo el análisis del rendimiento.
- **Medición de los resultados**, incluyendo los registros de pruebas.

5.9. Fase 6: Definición de Perfiles de Tráfico

Los perfiles de tráfico definen las condiciones de la red en el momento de llevarse a cabo la validación y la evaluación de la IEP. También recogen cuáles son las características del tráfico que se utiliza para llevar a cabo los análisis de los diferentes aspectos de rendimiento. Dado que las pruebas de conformidad y algunas pruebas de rendimiento no se ven afectadas por estas condiciones, únicamente en algunos conjuntos de pruebas de rendimiento se hace uso de estos perfiles.

Las características que deben aparecer recogidas en un perfil de tráfico son:

- **Protocolo**: Cuál es el protocolo o protocolos utilizados para el envío de datos. En el caso de ser varios protocolos, se informará de la distribución porcentual de los mensajes en dichos protocolos.
- **Tamaño de los paquetes de datos**
- **Sentido**: Puede ser del IEP al IDR, del IDR al IEP o bidireccional. En el caso de ser bidireccional, si el tráfico es asimétrico se indicará cuál es la proporción en cada sentido.
- **Medición**: Equipo o sistema que lleva a cabo la medición del parámetro de rendimiento.
- **Retardo**: Retardo artificial incluido en la red.

- **Pérdida de paquetes**: Porcentaje de pérdida de paquetes.
- **Reenvío de paquetes**: Porcentaje de paquetes reenviados.
- **Descripción**

Se definirán perfiles de tráfico que permitan la obtención de información del rendimiento a efectos comparativos entre las diferentes implementaciones (esto es, que permitan obtener información acerca del rendimiento máximo que la implementación es capaz de ofrecer), y perfiles que aporten información acerca del rendimiento de la implementación en condiciones más realistas en cuanto a tipo de tráfico a proteger y estado de la red durante su operación.

5.10. Fase 7: Otras Consideraciones

En esta fase se analizarán otras consideraciones que deban ser tenidas en cuenta a la hora de llevar a cabo la validación y evaluación de la IEP. Aspectos tales como recomendaciones de arquitecturas a utilizar, limitaciones de los conjuntos de pruebas propuestos, disponibilidad de la IDR, herramientas necesarias o recomendadas para la realización de las pruebas, etc.... deben ser analizados y documentados en esta fase de la metodología.

5.11. Fase 8: Tareas Finales

Para finalizar el proceso de análisis de la IEP, se llevará a cabo la recopilación de todo el material generado en fases anteriores, tanto referente al protocolo o arquitectura de seguridad, como las definiciones de los conjuntos de pruebas que deben aplicarse. Esta información se compilará en un formato que sea fácilmente accesible y manejable por los responsables de los análisis.

Se prestará especial atención a la utilización de herramientas necesarias para llevar a cabo el análisis de las implementaciones, con el fin de asegurar que las herramientas seleccionadas en la fase 1 de esta metodología fueron acertadas y no se han realizado modificaciones en ese aspecto.

Capítulo 6

Aplicación de la Metolodogía a la Arquitectura IPsec

6.1. Introducción

En este capítulo se procederá a aplicar la metodología de validación y evaluación de implementaciones de protocolos y arquitecturas de seguridad a la arquitectura de seguridad IPsec. Con el fin de solventar los problemas y necesidades expresados anteriormente, esta aplicación de la metodología definirá conjuntos de pruebas detallados de los que se a la evaluación de la conformidad con los diferentes estándares que definen la arquitectura de seguridad IPsec, y por otro a la obtención de información fiable acerca del rendimiento de la implementación estudiada.

Los motivos por los que se ha elegido IPsec sobre otros protocolos y arquitecturas de seguridad existentes, estandarizados y ampliamente extendidos son varios:

- **IPsec es la arquitectura de seguridad que se utilizará en la próxima generación de redes IP**, al encontrarse incluido en la versión 6 del protocolo IP (IPv6), lo que hace necesario el evaluar las capacidades de interoperabilidad y rendimiento de las actuales implementaciones, con el objeto de prevenir problemas de interoperatividad y/o rendimiento futuros.
- **IPsec es una arquitectura de seguridad compleja**, en la que se dan gran parte de los mecanismos de seguridad ofrecidos por los diferentes protocolos y arquitecturas, siendo posible la utilización de múltiples mecanismos y herramientas para llevar a cabo cada una de las funciones. Esto convierte a esta arquitectura en un examen exigente para la metodología, que servirá para medir hasta qué nivel la metodología es lo suficientemente completa como para abarcar todas

las posibilidades de IPsec, al tiempo que es lo suficientemente flexible como para permitir definir conjuntos de pruebas adaptados a cada protocolo y mecanismo presente en la arquitectura.

- **IPsec, junto con TLS, son los estándares de seguridad más extendidos en la actualidad**, al tiempo que las implementaciones de IPsec son las que más problemas de interoperabilidad entre ellas presentan. Por este motivo es necesario desarrollar un conjunto de pruebas que nos permitan conocer cuáles son las capacidades y limitaciones de una implementación determinada.

6.2. Conformidad con el estándar

Para las implementaciones de protocolos y arquitecturas de seguridad la conformidad con el estándar es un aspecto que debe ser evaluado de cara a poder disponer de información acerca del nivel de fidelidad en el cumplimiento de lo especificado en los documentos que definen los protocolos y herramientas criptográficas utilizadas. Esta información podrá ser utilizada posteriormente para evaluar la interoperatividad de la implementación estudiada con otras implementaciones de IPsec diferentes. En esta sección se definen las pruebas que es necesario llevar a cabo para poder evaluar el nivel de conformidad con el estándar de una implementación IPsec.

6.2.1. Requisitos

Para poder llevar a cabo las pruebas descritas en este documento, será necesario disponer de una implementación IPsec de referencia, conforme a los estándares de IPsec y sobre la que se disponga de control absoluto. Esta implementación de referencia debe poder ser capaz de desarrollar los protocolos IKE (e ISAKMP en el caso de IKEv1), ESP y AH para establecer túneles criptográficos, al tiempo que debe ser posible forzar a la implementación a enviar mensajes de cualquiera de los protocolos fuera de orden. Hay que resaltar que, dado que las especificaciones de IPsec de 1.998 (IPsec versión 1) y las de 2.005 (IPsec versión 2) no son compatibles entre sí, la implementación de referencia que utilicemos deberá utilizar la misma versión de la arquitectura de seguridad que la implementación que se desea evaluar.

Es también un requisito para la realización de las pruebas el disponer del control sobre la implementación que se validará, ya que será necesario realizar cambios en la configuración del establecimiento de túneles criptográficos tanto en la implementación de referencia como en la implementación a validar.

También es necesario disponer de mecanismos para generar las credenciales necesarias para validar los diferentes métodos de autenticación: una

infraestructura de clave pública con una autoridad de certificación que emita certificados X.509 y un generador de pares de claves RSA. En el caso de utilizar IPsec versión 2 y desear llevar a cabo la autenticación mediante EAP, los mecanismos necesarios para generar credenciales que puedan encapsularse en dicho protocolo también serán necesarios. En muchos casos podremos encontrarnos con que la propia implementación de IPsec que se desea validar puede desarrollar los mecanismos necesarios para generar certificados X.509, pares de claves RSA, etc.... Sin embargo, el uso de esos mecanismos para generar las credenciales no se recomienda, ya que existe la posibilidad de que incompatibilidades en dichas credenciales (por ejemplo, el no reconocer la autoridad de certificación raíz utilizada para firmar los certificados X.509) puedan propagarse hasta el proceso de autorización, generando falsos resultados.

6.2.2. Configuración de las pruebas

Para llevar a cabo todas las pruebas de conformidad es necesario disponer de una implementación de referencia con la que establecer los túneles criptográficos con la implementación evaluada. Cada una de estas implementaciones deberá configurarse para proteger una red en la que deberá existir al menos un equipo de usuario. Las direcciones de red asignadas deberán ser tales que no produzcan ningún conflicto con el resto de la infraestructura de red existente. En la metodología para análisis del rendimiento del IETF ([19]) se propone el uso del rango de direcciones desde 192.18.0.0 hasta 192.19.255.255, que ya ha sido reservado por el IANA para la aplicación de metodologías de rendimiento. Sin embargo, es posible que esas direcciones se encuentren en uso al estar llevando a cabo alguna medición del rendimiento de otros dispositivos de red, por lo que siempre es necesario comprobar la disponibilidad de las direcciones escogidas.

En las Figuras 6.2.2 y 6.2.2 se muestran las configuraciones necesarias para evaluar la conformidad de la implementación de IPsec cuando funciona en modo pasarela (Figura 6.2.2) y cuando funciona en modo de equipo final (Figura 6.2.2). En estas figuras se proponen también unas direcciones de red que serán las que se utilicen durante la elaboración de este conjunto de pruebas con el fin de eliminar posibles ambigüedades[1], por lo que al desarrollar la metodología podrán ser reemplazadas por los valores reales utilizados en la red en la que se lleve a cabo la validación. Hay que destacar que en ambos esquemas de red la implementación de referencia aparece representada funcionando en modo pasarela. Sin embargo, la utilización de implementaciones de referencia que funcionen en modo de equipo final no plantea problema alguno, por lo que puede aplicarse sin ningún tipo de problemas.

[1] Estas direcciones han sido escogidas de acuerdo a las directrices del IETF en [19]

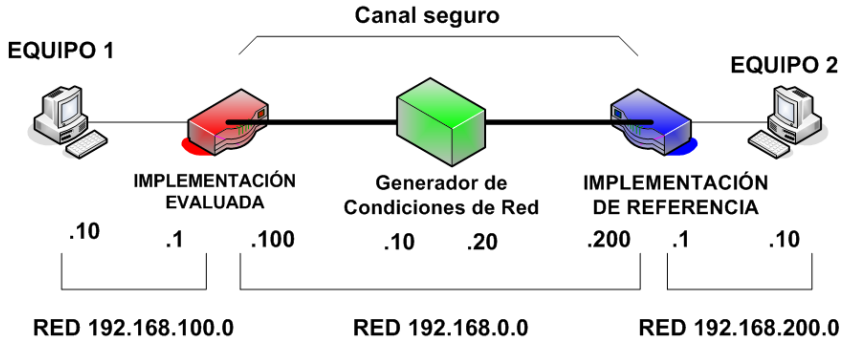

Figura 6.1: Esquema de red utilizado para la validación de una implementación de IPsec actuando como pasarela

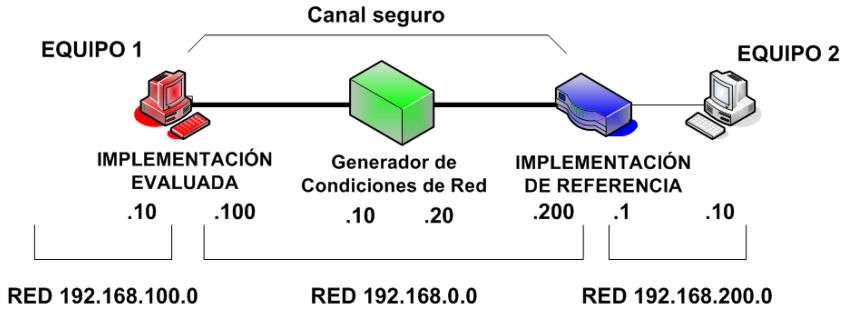

Figura 6.2: Esquema de red utilizado para la validación de una implementación de IPsec actuando como equipo final

Dado que el rendimiento no es el objeto de los análisis que se llevarán a cabo, resulta indiferente la infraestructura de red existente entre las dos implementaciones de IPsec, pudiendo existir tantos nodos intermedios como sea necesario.

Antes de comenzar las pruebas propuestas es necesario comprobar que la configuración de red de todos los dispositivos es correcta, por lo que, desactivando la utilización de IPsec, se comprobará que todos los dispositivos pueden establecer una comunicación con el resto de equipos involucrados en las pruebas mediante ICMP y los puertos UDP 500 (utilizado por ESP y AH) y UDP 4500 (utilizado al hacer uso de NAT traversal). Se recomienda prescindir de la utilización de NAT hasta llevar a cabo los análisis del resto de componentes de IPsec, siguiendo el orden establecido.

Por último, es también necesario evaluar las herramientas criptográficas de la implementación que utilicemos como referencia. Para ello podremos utilizar los vectores de pruebas existentes en la documentación de los diferentes algoritmos criptográficos.

6.2.3. Pruebas de conformidad criptográfica

6.2.3.1. Objetivos

Determinar la correcta implementación de las herramientas criptográficas utilizadas por IPsec y definidas en [70] y [60]. Aunque las recomendaciones y los algoritmos soportados se encuentran en constante evolución, las recomendaciones del IETF acerca de la conformidad con IPsec se restringen a un determinado conjunto de algoritmos y herramientas, que serán los validados de forma obligatoria.

6.2.3.2. Configuración necesaria

Las implementaciones de IPsec se configurarán para establecer los túneles criptográficos en modo túnel. También deberán utilizar el algoritmo NULL en la negociación de las Fases 1 y 2 de IKE[2]. En la Fase 1 de IKE se utilizará el modo acelerado. Para proteger el tráfico se utilizará el protocolo ESP. Para generar tráfico se enviará un mensaje "*Echo Request*" con el campo de datos "`AAAA`" hasta completar un tamaño mínimo de 512 bytes. La autenticación se establecerá a un secreto compartido seleccionable por el usuario.

La configuración de ESP para proteger el tráfico en la implementación a analizar consistirá en las siguientes suites criptográficas:

[2]En algunas implementaciones el algoritmo NULL no aparece como tal, sino que aparece como una opción para deshabilitar la confidencialidad o el cifrado en las Fases 1 y 2

- Algoritmo de cifrado NULL - Sin control de integridad
- 3DES en modo CBC y HMAC-SHA1-96
- AES con 128 bits de clave en modo CBC - AES-XCBC-MAC-96

Opcionalmente se podrán configurar otras suites criptográficas que sean de interés para el usuario.

Posteriormente, se configurará ESP para proteger el tráfico utilizando el algoritmo NULL y sin control de integridad, al tiempo que se configura la Fase 2 de IKE para que utilice las siguientes suites criptográficas:

- Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-MD5-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-MD5-96
- Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 2 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 2 (1024 bits), con AES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 2 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96

Finalmente, se restablecerá la configuración de la Fase 2 de IKE para que utilice 3DES y HMAC-SHA1-96, y se configurará la Fase 1 de IKE para que utilice las siguientes suites criptográficas:

- Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-MD5-96

- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-MD5-96
- Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 2 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 2 (1024 bits), con AES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 2 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96

Las configuraciones que utilizan HMAC-MD5-96 pertenecen en la actualidad al conjunto de configuraciones criptográficas que deben ser soportadas por las implementaciones de IPsec v1, pero que el propio IETF ha excluido de las recomendaciones de IPsec v2. Sin embargo, al encontrarnos en la actualidad en un periodo de transición entre ambas especificaciones se ha decidido incluir esta configuración entre las que es necesario evaluar.

Al igual que con ESP, el usuario tiene la potestad de validar otras suites criptográficas en cualquier fase de IKE utilizando el mismo procedimiento a continuación de estas pruebas obligatorias.

6.2.3.3. Procedimiento

Para cada una de las configuraciones reseñadas en el apartado anterior, el equipo 2 (con dirección 192.168.200.10) enviará al equipo 1 (192.168.100.10) mensajes ICMP "Echo Request" con las características comentadas anteriormente. Estos mensajes originarán la creación de un túnel criptográfico entre las implementaciones de IPsec y el envío de mensajes a través del mismo que generarán respuestas conocidas de antemano. Una vez enviados al menos diez de estos mensajes ICMP se procederá a comprobar las respuestas

(conforme a la evaluación que se indica en el apartado siguiente). Finalizado este proceso, se configurarán las implementaciones de IPsec con la siguiente configuración a evaluar y se repetirá el proceso.

Con el fin de agilizar el proceso, es posible configurar todas las configuraciones a probar como alternativas aceptables en la implementación a evaluar; de esta forma, tras finalizar las pruebas con una configuración únicamente será necesario modificar la implementación de referencia para que únicamente acepte la alternativa que deseamos evaluar.

En el caso de desear evaluar otras suites criptográficas, hay que tener en cuenta que:

- Cada grupo de Diffie-Hellman debe validarse con todos los algoritmos criptográficos y funciones resumen utilizados.
- Cada algoritmo criptográfico debe validarse con todos los grupos de Diffie-Hellman y funciones resumen utilizados.
- Cada función resumen debe validarse con todos los algoritmos criptográficos y grupos de Diffie-Hellman.

6.2.3.4. Mediciones y resultados

Para validar el resultado de cada prueba, será necesario analizar los mensajes "*Echo Reply*" que habrán llegado al equipo 2 procedentes del equipo 1. Para que un mensaje se considere correcto debe:

- Tener el mismo tamaño que el mensaje Echo Request enviado.
- Contener la misma información en el campo de datos.

En el caso de que se reciban los mensajes Echo Reply (tantos como mensajes Echo Request se enviasen) y todos ellos cumplan las dos condiciones anteriores, podemos constatar que la implementación evaluada es capaz de cifrar, descifrar, generar y validar resúmenes criptográficos correctamente con la suite criptográfica utilizada en la capa de IPsec que se configurase en cada momento.

En caso de que alguno de los mensajes Echo Reply presente algún problema será necesario almacenar la pareja de mensajes Echo Request y Echo Reply para llevar a cabo las labores de auditoría posterior.

6.2.3.5. Informe de resultados

Se informará de una implementación exitosa de una suite criptográfica concreta en una capa determinada de IPsec cuando todos los mensajes recibidos cumplan los requisitos determinados en el apartado anterior. En caso

contrario se informará de la cantidad de mensajes que presentaron errores y se informará del tipo de error encontrado.

6.2.4. Validación de protocolos y subprotocolos

6.2.4.1. Objetivos

Determinar la correcta implementación de los protocolos que forman parte de la arquitectura de seguridad IPsec, tal y como se define en [90] y [91] y sus documentos asociados. A pesar de la incompatibilidad entre ambas versiones de IPsec la aplicación de la metodología proporciona la forma de evaluar ambas ya que la funcionalidad y estructura es idéntica en ambas.

6.2.4.2. Configuración necesaria

La infraestructura necesaria para llevar a cabo la validación de los protocolos que componen la arquitectura de seguridad IPsec será la descrita en el apartado 6.2.2.

En cuanto a la configuración de las implementaciones IPsec, ambas deberán estar configuradas para llevar a cabo la autenticación mediante un secreto compartido escogido por el usuario, y deberán utilizar siempre, en todos los protocolos, el algoritmo de cifrado NULL sin control de integridad. La implementación de IPsec a evaluar deberá estar configurada de tal forma que no agrupe el tráfico similar en el mismo túnel criptográfico.

Las configuraciones de los diferentes protocolos seguirán el siguiente plan:

- Modo Túnel, Fase 1 de IKE con Main Mode, Fase 2 de IKE con Quick Mode, sin PFS, ESP
- Modo Túnel, Fase 1 de IKE con Main Mode, Fase 2 de IKE con Quick Mode, sin PFS, AH
- Modo Túnel, Fase 1 de IKE con Main Mode, Fase 2 de IKE con Quick Mode, con PFS, ESP
- Modo Túnel, Fase 1 de IKE con Main Mode, Fase 2 de IKE con Quick Mode, con PFS, AH
- Modo Túnel, Fase 1 de IKE con Modo Acelerado, Fase 2 de IKE con Quick Mode, sin PFS, ESP
- Modo Túnel, Fase 1 de IKE con Modo Acelerado, Fase 2 de IKE con Quick Mode, sin PFS, AH

- Modo Túnel, Fase 1 de IKE con Modo Acelerado, Fase 2 de IKE con Quick Mode, con PFS, ESP
- Modo Túnel, Fase 1 de IKE con Modo Acelerado, Fase 2 de IKE con Quick Mode, con PFS, AH
- Modo Transporte, Fase 1 de IKE con Main Mode, Fase 2 de IKE con Quick Mode, sin PFS, ESP
- Modo Transporte, Fase 1 de IKE con Main Mode, Fase 2 de IKE con Quick Mode, sin PFS, AH
- Modo Transporte, Fase 1 de IKE con Main Mode, Fase 2 de IKE con Quick Mode, con PFS, ESP
- Modo Transporte, Fase 1 de IKE con Main Mode, Fase 2 de IKE con Quick Mode, con PFS, AH
- Modo Transporte, Fase 1 de IKE con Modo Acelerado, Fase 2 de IKE con Quick Mode, sin PFS, ESP
- Modo Transporte, Fase 1 de IKE con Modo Acelerado, Fase 2 de IKE con Quick Mode, sin PFS, AH
- Modo Transporte, Fase 1 de IKE con Modo Acelerado, Fase 2 de IKE con Quick Mode, con PFS, ESP
- Modo Transporte, Fase 1 de IKE con Modo Acelerado, Fase 2 de IKE con Quick Mode, con PFS, AH

Para generar tráfico se enviará un mensaje "*Echo Request*" con el campo de datos "`AAAA`" hasta completar un tamaño mínimo de 2000 bytes. Al establecer nuevos túneles criptográficos se enviará desde el equipo origen un inicio de conexión TCP o UDP a puertos aleatorios del equipo destino.

6.2.4.3. Procedimiento

Para cada una de las configuraciones propuestas en el apartado anterior se procederá a establecer desde la implementación de referencia un túnel criptográfico con los parámetros definidos en la configuración. Una vez establecido el túnel criptográfico, se procederá a enviar tráfico ICMP desde el equipo 2 hacia el equipo 1, enviando un mínimo de 10 mensajes. Los mensajes de respuesta deberán cumplir que:

- Tienen el mismo tamaño que el mensaje Echo Request enviado.
- Contienen la misma información en el campo de datos.

A continuación se procederá a enviar los mensajes especialmente preparados para cada uno de los protocolos de IPsec, siguiendo la siguiente secuencia:

- Tienen el mismo tamaño que el mensaje Echo Request enviado.
- Contienen la misma información en el campo de datos.

Seguidamente se procederá a enviar mensajes especialmente formados desde la implementación de referencia para evaluar las respuestas de la implementación evaluada. Las pruebas que deben generar el rechazo de la implementación evaluada son:

- Solicitud de regeneración de las claves reutilizando el Nonce anterior.
- Solicitud de regeneración de las claves reutilizando el Vector de Inicialización anterior.
- Solicitud de configuración (CFG_REQ) inválido.
- Envío de un mensaje de repuesta de configuración (CFG_REP) sin mensaje de solicitud de configuración (CFG_REQ) previo.
- Envío de un identificador de equipo (FQDN) terminado en NULL.
- Envío de un identificador de equipo (FQDN) terminado en CR.
- Envío de un identificador de equipo (FQDN) terminado en CRLF.
- Negociación de suites criptográficas en las que aparece una longitud de clave para algoritmos con clave de tamaño fijo.
- Negociación de suites criptográficas en las que aparecen múltiples valores para un mismo atributo en la misma suite.
- Negociación de suites criptográficas en las que se propone un algoritmo de cifrado e integridad, y otro que únicamente ofrece integridad en la misma suite.
- Envío de un mensaje de IKE con un SPI con valor 0.
- Mensaje cifrado encapsulado en otro mensaje cifrado.
- Envío de un mensaje de notificación indicando un SPI inválido, cifrado dicho mensaje.
- Envío de un mensaje que obligue a generar una respuesta de notificación, y transmitir un nuevo mensaje en el que se altera el estado de la asociación de seguridad.

- Utilización de la clave de la Fase 1 de IKE en la Fase 2.
- Envío del secreto compartido durante la negociación terminado en NULL.
- Envío del secreto compartido de más de 64 bits durante la negociación.
- Envío de más tráfico del que soporta la implementación evaluada (indicado por la ventana de transmisión).
- Solicitud de respuesta a un mensaje informativo.
- Solicitud de cierre de un SPI y posterior reutilización de dicho SPI.
- Uso del bit crítico en una cabecera inválida de IKE.

Por su parte, los mensajes que deben ser aceptados por la implementación evaluada son los siguientes:

- Envío de una propuesta de configuración con ESP y AH en la misma propuesta (sólo uno de los protocolos aparecerá en la respuesta).
- Solicitud de renovación de una asociación de seguridad antes de tiempo y posterior envío de tráfico cuando la asociación de seguridad nueva y la antigua están activas.
- Codificación del secreto compartido durante la negociación en hexadecimal.

Finalmente, restableciendo la configuración inicial de las implementaciones de IPsec[3], se procederá a establecer un mínimo de 5 túneles criptográficos diferentes entre las implementaciones de IPsec. Para ello se generará tráfico desde el equipo 1 hacia el equipo 2 estableciendo conexiones a diferentes puertos TCP y UDP.

6.2.4.4. Mediciones y resultados

Para cada una de las configuraciones anteriores se almacenará todo el tráfico intercambiado entre las implementaciones de IPsec, así como los registros que pudieran haberse generado. Para cada mensaje especialmente formado se almacenará la respuesta de la implementación evaluada.

Para que una implementación pueda considerarse que implementa correctamente los protocolos de IPsec, el resultado de las pruebas debe ser

[3]Puede ser necesario reiniciar el equipo en caso de que se haya producido algún error durante las pruebas.

tal que se haya podido establecer el túnel criptográfico entre las implementaciones de IPsec con la configuración indicada, se hayan rechazado todos los mensajes especialmente creados definidos en el primer grupo del apartado anterior, y se han aceptado los mensajes definidos en el segundo grupo, adaptando el funcionamiento a lo indicado en dichos mensajes, y además, la implementación de IPsec evaluada ha sido capaz de separar el tráfico generado desde el equipo 1 hacia el equipo 2 en diferentes túneles criptográficos.

6.2.4.5. Informe de resultados

Se informará de una implementación exitosa de una configuración concreta cuando se cumplan las especificaciones determinadas en el apartado anterior. En caso contrario se informará de qué errores se han encontrado, indicando cuál ha sido el resultado obtenido y cuál es el resultado esperado.

6.2.5. Validación de los mecanismos de autenticación

6.2.5.1. Objetivos

Determinar la correcta implementación de los mecanismos de autenticación definidos en [65] y [25] y sus documentos asociados. En la siguiente relación de pruebas se indicará aquellos casos en los que alguna prueba únicamente deba aplicarse a una versión determinada de IPsec.

6.2.5.2. Configuración necesaria

La infraestructura necesaria para llevar a cabo la validación de los mecanismos de seguridad definidos para la arquitectura de seguridad IPsec será la descrita en el apartado 6.2.2.

En cuanto a la configuración de las implementaciones IPsec, ambas deberán estar configuradas para establecer túneles criptográficos utilizando ESP. Las suites criptográficas a utilizar dependerán de los resultados obtenidos en las pruebas de conformidad criptográfica, debiendo utilizarse alguna que haya pasado satisfactoriamente dichas pruebas.

Las configuraciones de los diferentes protocolos seguirán el siguiente plan:

- Autenticación únicamente mediante secreto compartido, protegido con MD5, utilizando Main Mode en la Fase 1 de IKE
- Autenticación únicamente mediante secreto compartido, protegido con MD5, utilizando Modo Acelerado en la Fase 1 de IKE
- Autenticación únicamente mediante secreto compartido, protegido con SHA-1, utilizando Main Mode en la Fase 1 de IKE

- Autenticación únicamente mediante secreto compartido, protegido con SHA-1, utilizando Modo Acelerado en la Fase 1 de IKE
- Autenticación únicamente mediante claves RSA, utilizando Main Mode en la Fase 1 de IKE
- Autenticación únicamente mediante claves RSA, utilizando Modo Acelerado en la Fase 1 de IKE
- Autenticación únicamente mediante certificados X.509, utilizando Main Mode en la Fase 1 de IKE
- Autenticación únicamente mediante certificados X.509, utilizando Modo Acelerado en la Fase 1 de IKE
- (Sólo para IKEv2) Autenticación únicamente mediante algún mecanismo encapsulado en EAP, utilizando Main Mode en la Fase 1 de IKE
- (Sólo para IKEv2) Autenticación únicamente mediante algún mecanismo encapsulado en EAP, utilizando Modo Acelerado en la Fase 1 de IKE

Para generar tráfico se enviará un mensaje "*Echo Request*" con el campo de datos "`AAAA`" hasta completar un tamaño mínimo de 512 bytes.

6.2.5.3. Procedimiento

Para cada configuración diferente descrita en el apartado anterior se procederá a enviar un mensaje desde el equipo 2 al equipo 1, lo que ocasionará el establecimiento del túnel criptográfico entre las dos implementaciones de IPsec, llevando a cabo la autenticación según el modo seleccionado. En esta primera prueba las credenciales de las dos implementaciones deberán permitir el establecimiento del túnel. Una vez establecido el túnel se procederá a enviar un mensaje Echo Request confirmando la recepción de la respuesta.

A continuación se cerrará el túnel criptográfico establecido y se modificarán las credenciales en la implementación de referencia, de tal forma que sea la implementación evaluada la que rechace el establecimiento del túnel criptográfico.

Finalmente, se restaurarán las credenciales en la implementación de referencia y se modificarán en la implementación evaluada, lo que llevará a la implementación de referencia a denegar el establecimiento del túnel.

Estas pruebas se llevarán a cabo para todas las configuraciones descritas en el apartado anterior.

6.2.5.4. Mediciones y resultados

Para cada una de las configuraciones anteriores se almacenará todo el tráfico intercambiado entre las implementaciones de IPsec, así como los registros de autenticación que se pudieran haber generado.

Para que una implementación pueda considerarse que implementa correctamente los métodos de autenticación de IPsec, el resultado de las pruebas debe ser tal que se haya podido establecer el túnel criptográfico entre las implementaciones de IPsec con la configuración indicada. Para que una implementación sea conforme con los métodos de autenticación de IKE bastará con que implemente correctamente los métodos de autenticación mediante secreto compartido y certificados X.509.

6.2.5.5. Informe de resultados

Se informará de una implementación exitosa de un método de autenticación concreto cuando el establecimiento de los túneles criptográficos haya sido exitoso en el caso de que las credenciales utilizadas fueran correctas y haya sido denegado en caso contrario. Se informará de una implementación conforme a IPsec si los métodos de autenticación mediante secreto compartido y certificados X.509 están implementados correctamente. La implementación incorrecta de otros métodos originará mensajes de aviso advirtiendo de la imposibilidad de utilizar dicho mecanismo.

6.2.6. Validación de la gestión de claves

6.2.6.1. Objetivos

Determinar la correcta implementación de los protocolos de gestión de claves de IPsec. Aunque los protocolos de gestión de claves pueden ser libremente implementados por los fabricantes (aunque la mayoría opta por utilizar el propuesto en los estándares: IKE), todos ellos deben contar con una serie de características que hagan posible su interoperatividad.

6.2.6.2. Configuración necesaria

La infraestructura necesaria para llevar a cabo la validación de los mecanismos de seguridad definidos para la arquitectura de seguridad IPsec será la descrita en el apartado 6.2.2.

En cuanto a la configuración de las implementaciones IPsec, ambas deberán estar configuradas para establecer túneles criptográficos utilizando ESP. Las suites criptográficas a utilizar dependerán de los resultados obtenidos en las pruebas de conformidad criptográfica, debiendo utilizarse alguna que haya pasado satisfactoriamente dichas pruebas.

En el caso de utilizar IPsec v1 las implementaciones necesitar negociar los parámetros de configuración, por lo que la secuencia de configuraciones necesarias será la siguiente para ambas implementaciones:

- Caducidad de la clave de la Fase 1 de IKE en 30 segundos.
- Caducidad de la clave de la Fase 2 de IKE en 30 segundos.
- Caducidad de la clave de la Fase 1 de IKE al transmitir 10 KBytes.
- Caducidad de la clave de la Fase 2 de IKE al transmitir 10 KBytes.

Sin embargo, si se utiliza IPsec v2 las configuraciones que serán utilizadas son[4]:

- Caducidad de la clave de la Fase 1 de IKE en 30 segundos en la implementación de referencia.
- Caducidad de la clave de la Fase 2 de IKE en 30 segundos en la implementación de referencia.
- Caducidad de la clave de la Fase 1 de IKE al transmitir 10 KBytes en la implementación de referencia.
- Caducidad de la clave de la Fase 2 de IKE al transmitir 10 KBytes en la implementación de referencia.
- Caducidad de la clave de la Fase 1 de IKE en 30 segundos en la implementación evaluada.
- Caducidad de la clave de la Fase 2 de IKE en 30 segundos en la implementación evaluada.
- Caducidad de la clave de la Fase 1 de IKE al transmitir 10 KBytes en la implementación evaluada.
- Caducidad de la clave de la Fase 2 de IKE al transmitir 10 KBytes en la implementación evaluada.

Para generar tráfico se enviará un mensaje "*Echo Request*" con el campo de datos "`AAAA`" hasta completar un tamaño mínimo de 2000 bytes.

[4]Para cada una de estas configuraciones la implementación de la que no se dice nada tiene valores mayores de caducidad de las claves

6.2.6.3. Procedimiento

Para cada configuración diferente descrita en el apartado anterior se procederá a establecer un túnel criptográfico entre las implementaciones de IPsec, comprobando si el proceso de renegociación de claves se lleva a cabo de forma satisfactoria. Debido a las características de la caducidad de las claves en IKEv1 y en IKEv2, si estamos evaluando una implementación de IKEv1 basta con evaluar el caso en que ambas implementaciones acuerdan renovar las claves, mientras que en IKEv2 es necesario evaluar el caso en el que cada implementación necesita renegociar las claves.

Para las configuraciones en las que la caducidad se define en función del tráfico protegido, una vez establecido el túnel criptográfico se procederá a enviar información desde el equipo 2 al equipo 1, hasta forzar el proceso de renegociación de claves.

Estas pruebas se llevarán a cabo para todas las configuraciones descritas en el apartado anterior.

6.2.6.4. Mediciones y resultados

Para cada una de las configuraciones anteriores se almacenará todo el tráfico intercambiado entre las implementaciones de IPsec, así como los registros de log que pudieran haberse generado.

Para que una implementación pueda considerarse que implementa correctamente la gestión de claves criptográficas tal y como se describe en la arquitectura de seguridad IPsec, el resultado de las pruebas debe ser tal que se hayan realizado todos los procesos de renegociación de claves de forma satisfactoria; esto es, sin producirse errores que interrumpan una posible comunicación entre las redes protegidas.

6.2.6.5. Informe de resultados

Se informará de una implementación exitosa de la gestión de claves en IPsec cuando en todas las configuraciones anteriores la implementación evaluada haya realizado exitosamente la gestión de claves.

6.2.7. Validación de otras características

6.2.7.1. Objetivos

Determinar la correcta implementación de características incluidas en los estándares que conforman la arquitectura de seguridad de IPsec y que son necesarias para establecer túneles criptográficos en determinadas topologías de red o situaciones concretas. Estas características son la compresión de

datos a nivel IP mediante IPComp (definido en [140]), el NAT traversal (definido en [65] y [25]) y la notificación explícita de congestión en la red (definida en [132] e incorporada a la arquitectura IPsec en [25]).

6.2.7.2. Configuración necesaria

La infraestructura necesaria para llevar a cabo la validación de los mecanismos de seguridad definidos para la arquitectura de seguridad IPsec será la descrita en el apartado 6.2.2. Para poder evaluar la utilización de NAT traversal será necesario que las dos implementaciones de IPsec realicen NAT sobre la red que protegen.

En cuanto a la configuración de las implementaciones IPsec, ambas deberán estar configuradas para establecer túneles criptográficos utilizando ESP. Las suites criptográficas a utilizar en IKE dependerán de los resultados obtenidos en las pruebas de conformidad criptográfica, debiendo utilizarse alguna que haya pasado satisfactoriamente dichas pruebas. En cuanto a ESP se recomienda utilizar el algoritmo NULL para proteger la información, aunque la utilización de otros algoritmos no alteraría el resultado de estas pruebas.

Para poder llevar a cabo las pruebas de compresión de datos en la capa IP será necesario que ambas implementaciones de seguridad incluyan en su configuración la necesidad de utilizar dicha característica.

Para generar el tráfico necesario para la evaluación de la notificación de la congestión de red se establecerán dos conexiones UDP entre los equipos 1 y 2 (una en cada sentido) y se procederá a transmitir información tan rápidamente como sea posible. Para el resto de pruebas la comunicación se establecerá enviando mensajes "*Echo Request*" con el campo de datos "`AAAA`" hasta completar un tamaño mínimo de 2000 bytes.

6.2.7.3. Procedimiento

Para evaluar el funcionamiento de la implementación de NAT traversal se establecerán dos túneles criptográficos entre los equipos 1 y 2 (uno en cada sentido), para cada una de las siguientes configuraciones:

- La implementación evaluada utiliza NAT traversal; la implementación de referencia no utiliza NAT traversal.
- La implementación evaluada no utiliza NAT traversal; la implementación de referencia utiliza NAT traversal.
- La implementación evaluada utiliza NAT traversal; la implementación de referencia utiliza NAT traversal.

Tras el establecimiento de cada par de túneles criptográficos se transmitirá información por dichos túneles hasta enviar un mínimo de 5 mensajes en cada sentido.

La evaluación de la compresión de datos en la capa IP se llevará a cabo habilitando esta opción en ambas implementaciones. Una vez establecido un túnel criptográfico en estas condiciones se procederá a enviar un mínimo de 5 mensajes desde el equipo 1 hacia el equipo 2.

Por último, para llevar a cabo la evaluación de la notificación explícita de congestión, se establecerá un túnel criptográfico entre ambas implementaciones y se procederá a enviar tráfico tal y como se describe en el apartado anterior durante un mínimo de 30 segundos.

6.2.7.4. Mediciones y resultados

Al llevarse a cabo las pruebas para la evaluación de la implementación del NAT traversal se almacenarán los posibles registros de log que se hayan generado, así como el resultado del establecimiento de la comunicación entre los equipos 1 y 2 y los mensajes ICMP de respuesta que hayan recibido los equipos 1 y 2. Para que una implementación desarrolle correctamente el NAT traversal los mensajes deben haber sido respondidos en ambos sentidos sin errores y en las tres configuraciones propuestas en el apartado anterior.

En cuanto a la compresión en la capa IP, al llevarse a cabo esta prueba se almacenarán los registros de log que se hayan generado durante la ejecución de la misma y los mensajes ICMP de respuesta recibidos por el equipo 1. La implementación evaluada desarrollará correctamente esta técnica de compresión si los mensajes ICMP recibidos se corresponden con mensajes ICMP de Echo Reply correspondientes a los enviados por ese mismo equipo.

Por último, la evaluación de la notificación explícita de la congestión de la red requiere del almacenamiento de los registros de log producidos por las implementaciones de IPsec, así como los de otros dispositivos de red que pudieran existir en la infraestructura de pruebas y los mensajes ICMP que otros equipos hayan podido generar al detectar la congestión de la red. La implementación evaluada necesitará haber enviado los mensajes de notificación de la congestión y haber operado en consecuencia (por ejemplo, reduciendo su ventana de transmisión) para poder ser considerada como conforme al estándar.

6.2.7.5. Informe de resultados

Se informará de una implementación exitosa de NAT traversal cuando en todas las configuraciones anteriores la implementación evaluada haya sido capaz de redirigir el tráfico a través del NAT al equipo de destino correcto, sin pérdida de datos.

Se informará de un desarrollo exitoso de la compresión de datos en la capa IP cuando la implementación estudiada haya podido negociar y utilizar dicha característica al establecer canales seguros con la implementación de referencia.

Se informará de una implementación exitosa de la notificación explícita de congestión en la red cuando la implementación haya enviado y recibido los mensajes de notificación de la congestión al incrementar el tráfico en la red, y haya modificado sus parámetros de envío de datos para adaptarse a la nueva situación.

6.3. Evaluación del rendimiento

El rendimiento de una implementación de un protocolo de seguridad es un aspecto fundamental para poder conocer y planificar la evolución futura de nuestra red de comunicaciones. De esta forma, además de adquirir un conocimiento acerca de nuestra infraestructura que es indispensable de cara a prevenir ataques que pudieran impedir el desarrollo de las actividades para las que estaba prevista dicha red (por ejemplo, los ataques de Denegación de Servicio), estamos facilitando el identificar, promover y utilizar las configuraciones, técnicas y herramientas que mejor rendimiento pueden ofrecernos; es decir, estamos asegurándonos de utilizar la implementación de IPsec de forma óptima. En esta sección se definen las pruebas que es necesario llevar a cabo para poder evaluar correctamente el rendimiento que puede ofrecer nuestra implementación de IPsec.

6.3.1. Requisitos

Las pruebas están orientadas a la evaluación del rendimiento de la implementación de IPsec, por lo que es necesario que la infraestructura de red sobre la que se realizan las pruebas cuente con las siguientes características:

- Sea una arquitectura dedicada, de forma que no haya otros equipos o dispositivos que puedan interferir en la medición de resultados[5].
- Tenga una capacidad mayor que la máxima del dispositivo que se evalúa. Al establecer cuál es la máxima capacidad de tráfico se habrá de considerar que el dispositivo puede utilizar el máximo ancho de banda en modo Full-Duplex para cada una de sus conexiones de red por las que establezcan túneles IPsec.

Adicionalmente, será necesario disponer de suficientes equipos en cada una de las redes protegidas (la red protegida por la implementación a evaluar

[5]Como ya se discutió en el capítulo 5, es posible utilizar una arquitectura de red no dedicada, pero los resultados que se obtendrán no serán precisos

y la protegida por la implementación de referencia descrita en el apartado 6.2.1) para poder saturar la red que unirá ambas implementaciones.

Dado que es necesario modificar la configuración de las suites criptográficas utilizadas en el establecimiento de túneles criptográficos y la protección de la información, es preciso disponer del control sobre la implementación que se validará. Esta configuración se centrará sobre todo en las herramientas criptográficas que se utilizarán, por lo que el haber llevado a cabo la validación de la conformidad con el estándar previamente es también una necesidad.

En cuanto a la medición y generación del tráfico que se utilizará, será necesario contar con software o hardware capaz de establecer comunicaciones ICMP, TCP y UDP entre diferentes equipos, enviando datos tanto a una velocidad determinada (por ejemplo, 3 Mbps) como sin límite en la velocidad de transferencia. También será preciso que este software pueda iniciar conexiones nuevas entre equipos a intervalos regulares de tiempo, mientras mantiene las conexiones establecidas de antemano y envía información por dichas conexiones. En cuanto a la medición, será necesario disponer de herramientas similares a IPtraf ([80]) en ambas redes y en la red que une las implementaciones de IPsec.

6.3.2. Configuración de las pruebas

Con el fin de ejecutar las pruebas de rendimiento que se describirán a continuación será necesario disponer de una implementación de IPsec con la que la implementación de IPsec analizada pueda establecer túneles criptográficos, tal y como se ha descrito en el apartado anterior. Adicionalmente, es necesario que ambas implementaciones protejan una red en la que deberán existir al menos tantos equipos como sea preciso para saturar los enlaces de red que comunican la implementación de referencia con la implementación evaluada. Por ejemplo, si las implementaciones de IPsec están unidas por dos enlaces Fast Ethernet de 100 Mbps, será necesario que en cada una de las redes existan equipos suficientes para generar un tráfico de, al menos, 200 Mbps en dirección a la otra red. Antes de comenzar a realizar las pruebas de rendimiento es aconsejable verificar que los equipos, sin utilizar IPsec, pueden utilizar el máximo ancho de banda posible en la infraestructura de red utilizada (en el caso anterior, debería poderse obtener un ancho de banda acumulado de alrededor de 400 Mbps, resultante de obtener el máximo ancho de banda de la red (100 Mbps) por cada enlace físico (2 enlaces) y en cada sentido (al utilizar Full-Duplex es posible obtener el máximo ancho de banda de Ethernet en cada sentido de la comunicación)).

Entre ambas redes se situará un equipo de red que sea capaz de alterar las condiciones de la red (retardo, pérdida y reenvío de paquetes) de forma artificial, lo que nos permitirá comprobar cuál es el comportamiento de la

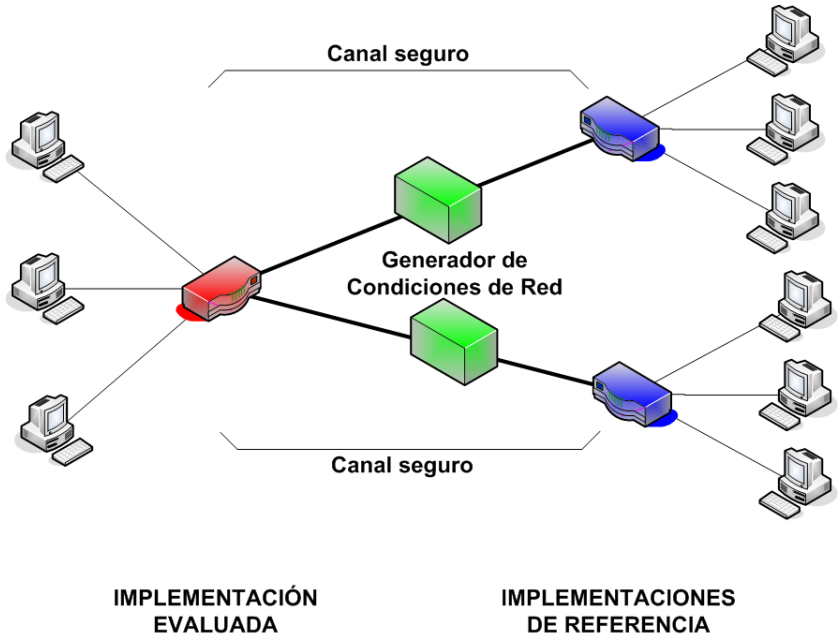

Figura 6.3: Posible esquema de red utilizado para la evaluación del rendimiento de una implementación IPsec.

implementación IPsec en determinadas condiciones de la red. Este equipo de red debe tener capacidad de proceso suficiente para distribuir todo el tráfico que deberá enviarse por la red en ambos sentidos. Una posible solución es la utilización de un ordenador personal con software de emulación de condiciones de red, como por ejemplo [124].

Al igual que ocurría anteriormente, las direcciones de red asignadas deberán ser tales que no produzcan ningún conflicto con el resto de la infraestructura de red existente. En este caso se recomienda utilizar una infraestructura de red dedicada, por lo que las direcciones de red utilizadas no deberían afectar a las pruebas; sin embargo, si esto no es posible, se recomienda utilizar el rango reservado por el IANA para este tipo de pruebas (desde 192.18.0.0 hasta 192.19.255.255).

Por lo tanto, una posible infraestructura de red en este caso sería la representada en la figura 6.3.2. En el caso de que la implementación evaluada sólo pueda operar en modo de equipo final, la configuración se simplifica, ya que los máximos anchos de banda posibles son menores. Sin embargo, todas las pruebas pueden ser llevadas a cabo igualmente, por lo que no se especificará si las pruebas se llevan a cabo en una configuración o en otra.

Antes de comenzar las pruebas propuestas es necesario comprobar que la configuración de red de todos los dispositivos es correcta, por lo que,

desactivando la utilización de IPsec, se comprobará que todos los dispositivos pueden establecer una comunicación con el resto de equipos involucrados en las pruebas mediante ICMP y un rango de al menos 1000 puertos TCP y otros 1000 UDP.

En cada uno de los equipos instalados en las redes protegidas por las implementaciones de IPsec se implantará un medidor del tráfico de red enviado y recibido y un generador de tráfico de red con las características descritas anteriormente.

6.3.3. Perfiles de tráfico

Para llevar a cabo mediciones de rendimiento realistas y alejadas de máximos teóricos que no se corresponden con situaciones reales, se definen perfiles de tráfico para obtener información de rendimiento en función del tipo de uso que se haga de la red de comunicaciones. Los siguientes perfiles de tráfico serán obligatorios en aquellas pruebas en las que sea necesario utilizar perfiles de tráfico. Queda a voluntad del usuario el incluir otros perfiles en el conjunto de aquellos que se utilizarán en las pruebas.

Perfil: Máximo ancho de banda

Protocolo UDP

Tamaño de los paquetes de datos Igual al MTU

Sentido Bidireccional (ambas entidades se envían tráfico)

Medición En el receptor. El ancho de banda total será la suma del tráfico en los dos receptores.

Retardo artificial 0ms

Pérdida de paquetes 0 %

Reenvío de paquetes 0 %

Descripción En este perfil las entidades involucradas se envían mensajes utilizando el protocolo UDP sin esperar ni generar ninguna respuesta. Dado que no se garantiza que todo el tráfico UDP que se envía llegue a su destino, la medición del tráfico que ha sido transmitido por el canal IPsec se realizará en el destino.

Perfil: Ancho de banda en red saturada

Protocolo UDP

Tamaño de los paquetes de datos Igual al MTU

Sentido Bidireccional (ambas entidades se envían tráfico)

Medición En el receptor. El ancho de banda total será la suma del tráfico en los dos receptores.

Retardo artificial 1500ms

Pérdida de paquetes 40 %

Reenvío de paquetes 65 %

Descripción En este perfil las entidades involucradas se envían mensajes utilizando el protocolo UDP sin esperar ni generar ninguna respuesta. Dado que no se garantiza que todo el tráfico UDP que se envía llegue a su destino, la medición del tráfico que ha sido transmitido por el canal IPsec se realizará en el destino. Las condiciones de red se modifican artificialmente para obtener información del comportamiento en una red saturada.

Perfil: Usuario Tradicional

Protocolo 90 % TCP, 9 % UDP, 1 % ICMP

Tamaño de los paquetes de datos 250 bytes

Sentido Unidireccional (una entidad envía tráfico; la otra sólo envía ACKs). Los equipos situados en la red protegida por la implementación de IPsec analizada realizan el papel de clientes de estas comunicaciones.

Medición En el receptor, debido a la presencia de tráfico UDP.

Retardo artificial 0ms

Pérdida de paquetes 1 %

Reenvío de paquetes 2 %

Descripción Este perfil simula un comportamiento de tráfico similar al de la mayoría de los usuarios de redes de ordenadores, utilizando protocolos basados en TCP de forma mayoritaria, y dejando únicamente un pequeño margen para el tráfico UDP e ICMP.

Perfil: TCP Asimétrico Entrante

Protocolo TCP

Tamaño de los paquetes de datos Igual al MTU

Sentido Unidireccional (una entidad envía tráfico; la otra sólo envía ACKs). Los equipos situados en la red protegida por la implementación de IPsec analizada realizan el papel de servidores de esta comunicación.

Medición En el emisor o en el receptor.

Retardo artificial 0ms

Pérdida de paquetes 1 %

Reenvío de paquetes 2 %

Descripción Este perfil modela la utilización de protocolos basados en TCP de forma mayoritaria. Su interés radica en la elevada desigualdad entre el nivel de tráfico enviado y recibido.

Perfil: TCP Asimétrico Saliente

Protocolo TCP

Tamaño de los paquetes de datos Igual al MTU

Sentido Unidireccional (una entidad envía tráfico; la otra sólo envía ACKs). Los equipos situados en la red protegida por la implementación de IPsec analizada realizan el papel de clientes de esta comunicación.

Medición En el emisor o en el receptor.

Retardo artificial 0ms

Pérdida de paquetes 1 %

Reenvío de paquetes 2 %

Descripción Este perfil modela la utilización de protocolos basados en TCP de forma mayoritaria. Su interés radica en la elevada desigualdad entre el nivel de tráfico enviado y recibido.

Perfil: TCP Simétrico

Protocolo TCP

Tamaño de los paquetes de datos Igual al MTU

Sentido Bidireccional (ambas entidades se envían tráfico)

Medición En el emisor o en el receptor.

Retardo artificial 0ms

Pérdida de paquetes 1 %

Reenvío de paquetes 2 %

Descripción Este perfil sirve para obtener información acerca del máximo rendimiento que la implementación podrá ofrecer al utilizar aplicaciones basadas en TCP. El rendimiento de este perfil será menor que el del perfil de máximo rendimiento ya que la necesidad de esperar la confirmación de la entrega de los mensajes ralentiza la comunicación.

Perfil: TCP Simétrico en red saturada

Protocolo TCP

Tamaño de los paquetes de datos Igual al MTU

Sentido Bidireccional (ambas entidades se envían tráfico)

Medición En el emisor o en el receptor.

Retardo artificial 1500ms

Pérdida de paquetes 40 %

Reenvío de paquetes 65 %

Descripción Este perfil sirve para obtener información acerca del máximo rendimiento que la implementación podrá ofrecer al utilizar aplicaciones basadas en TCP. La saturación artificial de la red nos indicará cuál es el comportamiento de la implementación en situaciones no óptimas.

Para generar artificialmente las condiciones de red descritas en los perfiles, se recomienda la utilización de hardware especializado, o herramientas como NIST Net ([124]), que permiten introducir alteraciones en el comportamiento de la red con gran flexibilidad y eficacia.

6.3.4. Ancho de banda

6.3.4.1. Objetivos

Determinar el máximo ancho de banda que la implementación de IPsec puede proteger cuando la información es transmitida de acuerdo a los patrones de tráfico definidos en el apartado anterior y a otros en los que el usuario pueda estar interesado. Nótese que en los perfiles se hace referencia al tamaño de los paquetes que se envían, aunque realmente nos referimos al tamaño de los mensajes que se transmiten, sean del nivel que sean.

6.3.4.2. Configuración necesaria

Las implementaciones de IPsec se configurarán para establecer los túneles criptográficos utilizando en las dos fases de IKE cualquier suite criptográfica y mecanismo de autenticación que permita la interoperatividad entre ellas.

La configuración de los protocolos ESP y AH para proteger el tráfico en la implementación a analizar consistirá en las siguientes suites criptográficas:

- ESP utilizando el algoritmo de cifrado NULL - Sin control de integridad
- ESP utilizando 3DES en modo CBC y HMAC-MD5-96
- ESP utilizando 3DES en modo CBC y HMAC-SHA1-96
- ESP utilizando AES con 128 bits de clave en modo CBC - AES-XCBC-MAC-96
- AH utilizando HMAC-MD5-96
- AH utilizando HMAC-SHA1-96
- AH utilizando AES-XCBC-MAC-96

Opcionalmente se podrán configurar otras suites criptográficas que sean de interés para el usuario.

En los equipos situados en las redes protegidas por las implementaciones de IPsec se configurarán los mecanismos de medición del tráfico emitido y recibido por cada equipo. Asimismo, se configurarán también los mecanismos de generación del tráfico para utilizar, al menos, los siguientes perfiles:

- Máximo ancho de banda
- Usuario Tradicional

- TCP Asimétrico entrante
- TCP Asimétrico saliente
- TCP Simétrico

Tras realizar los análisis con estos perfiles de tráfico se podrán utilizar otros en los que el usuario esté interesado.

El uso de compresión en la capa de datos IP no afecta a la medida del ancho de banda, aunque se desaconseja para evitar efectos colaterales en la implementación analizada

6.3.4.3. Procedimiento

Para cada una de las configuraciones de ESP reseñadas en el apartado anterior, se generará tráfico de acuerdo con cada uno de los perfiles de tráfico que se utilicen, utilizando tantos equipos como sea necesario. Una vez iniciada la generación de tráfico, se mantendrá dicha generación durante al menos 10 segundos, para facilitar que las velocidades de transmisión se estabilicen. Pasado este tiempo de estabilización, se iniciará la medición del ancho de banda utilizado, de acuerdo con lo especificado en el perfil. Dicha medición se llevará a cabo durante al menos 5 segundos, almacenando el ancho de banda medio que se ha utilizado durante este periodo de medición. Este proceso se repetirá 3 veces, calculando finalmente el valor medio de esas tres medidas.

El procedimiento descrito se llevará a cabo para cada perfil de tráfico en cada configuración, por lo que el mínimo número de medidas a realizar será de 75 medidas. Hay que tener en cuenta que tras sobrecargar la red puede ser necesario reiniciar los dispositivos o, al menos, cerrar todas las asociaciones de seguridad y esperar un periodo entre 15 y 30 segundos para que la implementación libere recursos ocupados.

Si se decide utilizar otros perfiles de tráfico, es necesario recordar que cada perfil de tráfico debe ser evaluado utilizando todas las configuraciones de ESP y AH posibles.

6.3.4.4. Mediciones y resultados

La medición del ancho de banda utilizado en cada túnel criptográfico se hará de acuerdo a lo especificado en cada perfil. En caso de duda, hay que tener en cuenta que el tráfico de los protocolos que no aseguran la entrega del tráfico cursado ha de ser medido en el receptor de dicho tráfico, ya que no hay garantías de que todos los mensajes enviados por el origen hayan sido transmitidos a través del túnel IPsec. El tráfico de protocolos que sí incluyen

mecanismos para asegurar la entrega de los mensajes puede ser medido en el origen o en el destinatario indistintamente.

La medición debe llevarse a cabo tras haberse estabilizado las velocidades de transmisión. Normalmente 10 segundos bastan para ello, aunque si al iniciar la medición se detectase una alta variabilidad en los anchos de banda que se transmiten, se deberá esperar un periodo adicional de seguridad. Al llevar a cabo la medida habrá que llevar a cabo la medición del ancho de banda durante un periodo mínimo de 5 segundos, almacenándose los valores medio, máximo y mínimo de ese periodo. Este proceso se repetirá tres veces, calculando la media aritmética de los valores medios, y almacenando los valores máximos y mínimos totales.

Al finalizar todas las medidas se dispondrá de información acerca del ancho de banda máximo que podemos obtener para cada perfil de tráfico, según cada una de las configuraciones de los protocolos ESP y AH.

6.3.4.5. Informe de resultados

Al finalizar las pruebas se informará al usuario de los resultados obtenidos, informando para cada par "Perfil - Configuración de ESP / AH" de:

- Valor medio del ancho de banda medido.[6].
- Valor máximo del ancho de banda medido.
- Valor mínimo del ancho de banda medido.

6.3.5. Máximo número de AS simultáneas

6.3.5.1. Objetivos

Determinar la cantidad máxima de asociaciones de seguridad que la implementación de IPsec analizada puede mantener negociadas simultáneamente, sin que se transmita información alguna por dichas asociaciones. El establecimiento de las asociaciones de seguridad se llevará a cabo de acuerdo a lo especificado en [65], [101] y [25]. La medición de este parámetro se llevará a cabo basándose en la propuesta similar del IETF para la medición del rendimiento de cortafuegos, especificada en [69].

[6]Este valor medio representa el Ancho de Banda máximo que podemos esperar para este perfil de tráfico con esta configuración, ya que no representa un extremo estadístico o *outlier*

6.3.5.2. Configuración necesaria

Las implementaciones de IPsec se configurarán para establecer los túneles criptográficos utilizando en las dos fases de IKE cualquier suite criptográfica y mecanismo de autenticación que permita la interoperatividad entre ellas, ya que el rendimiento de los mecanismos utilizados no afectará a los resultados de las pruebas. Se configurará el protocolos AH para proteger el tráfico, utilizando cualquier mecanismo disponible. Se configurarán las bases de datos de políticas de IPsec de tal forma que cada conexión se proteja mediante un túnel criptográfico propio.

En los equipos situados en las redes protegidas por las implementaciones de IPsec se configurarán los mecanismos para establecer conexiones TCP entre dos máquinas, sin intercambio de información alguno por dichas conexiones. En las redes protegidas por la(s) implementación(es) de IPsec de referencia se situarán los clientes, mientras que en red protegida por la implementación evaluada se situarán los servidores. Estos mecanismos debes ser capaces de iniciar una nueva conexión pasado un determinado tiempo, mientras mantiene el resto de conexiones abiertas. Además, estos mecanismos alertarán en el caso de que una conexión establecida se haya cerrado, o si un intento de conexión es infructuoso.

El uso de compresión en la capa de datos IP no afecta a la medida del número máximo de asociaciones de seguridad simultáneas, aunque se desaconseja para evitar efectos colaterales en la implementación analizada

6.3.5.3. Procedimiento

Una vez configurados los dispositivos y la infraestructura de red, se procederá a establecer conexiones entre los dispositivos protegidos por las implementaciones IPsec de referencia y la implementación analizada, a razón de una nueva conexión por segundo[7]. Este proceso continuará hasta que se de una de las siguiente condiciones:

1. Un intento de establecimiento de conexión resulta fallido.
2. Una conexión previamente establecida se cierra.

En este momento se almacenará la cantidad de asociaciones de seguridad establecidas hasta el momento en que se ha dado la situación que ha interrumpido el proceso, y se cerrarán todos los túneles criptográficos que permanezcan abiertos.

Se repetirá este mismo proceso incrementando a 2 segundos el periodo entre conexiones, con el fin de descartar saturaciones de la CPU de la imple-

[7]Este tiempo empezar a medirse a partir de la finalización del establecimiento de conexión anterior, para evitar el solapamiento de establecimientos de conexión.

mentación. En el caso de que la cantidad de asociaciones de seguridad sea la misma que en caso anterior, se finalizará la prueba; en caso contrario, se repetirá el proceso incrementando el periodo entre conexiones. Únicamente cuando dos procesos de establecimiento de conexiones con diferentes intervalos entre conexiones ofrezcan los mismos resultados se finalizará el desarrollo de estas pruebas.

En caso de estimar que el valor inicial de un segundo entre cada establecimiento de conexión es demasiado alto o bajo para la implementación que se está analizando, es posible utilizar otro valor que se aproxime más al valor con el que determinaremos el número máximo de asociaciones de seguridad.

6.3.5.4. Mediciones y resultados

En el momento en que el proceso de establecimiento de conexiones se interrumpa por alguna de las condiciones indicadas en el apartado anterior, se almacenará el número de asociaciones de seguridad establecidas previamente. Al confirmar que la cantidad de asociaciones de seguridad establecidas es el máximo para la implementación estudiada (según se describe en el apartado anterior), se presentará ese valor como resultado de las pruebas realizadas.

6.3.5.5. Informe de resultados

El informe de resultados de estas pruebas indicará, para todas las iteraciones en el proceso de establecimiento de conexiones llevadas a cabo, cuál ha sido su intervalo entre establecimiento de conexiones y cuántas asociaciones de seguridad se han llegado a establecer.

6.3.6. Capacidad de establecimiento de asociaciones de seguridad

6.3.6.1. Objetivos

Determinar la cantidad máxima de asociaciones de seguridad que la implementación de IPsec puede establecer por unidad de tiempo, mientras cada asociación de seguridad protege un determinado ancho de banda. El establecimiento de asociaciones de seguridad mientras otros túneles criptográficos previamente establecidos hacen uso de parte del ancho de banda disponible para transmitir información es una situación muy normal hoy en día y que puede ser origen de ataques de denegación de servicio.

6.3.6.2. Configuración necesaria

Las implementaciones de IPsec se configurarán para establecer los túneles criptográficos de acuerdo a las siguientes configuraciones para IKE y ESP / AH:

- IKE Fase 1
  - Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-MD5-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-MD5-96
  - Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-SHA1-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-SHA1-96
  - Grupo de Diffie-Hellman 2 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
  - Grupo de Diffie-Hellman 2 (1024 bits), con AES en modo CBC y HMAC-SHA1-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y HMAC-SHA1-96
  - Grupo de Diffie-Hellman 2 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96
- IKE Fase 2
  - Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-MD5-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-MD5-96
  - Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-SHA1-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-SHA1-96
  - Grupo de Diffie-Hellman 2 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96

  - Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
  - Grupo de Diffie-Hellman 2 (1024 bits), con AES en modo CBC y HMAC-SHA1-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y HMAC-SHA1-96
  - Grupo de Diffie-Hellman 2 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96
- ESP / AH
  - ESP con Algoritmo de cifrado NULL - Sin control de integridad
  - ESP con 3DES en modo CBC y HMAC-SHA1-96
  - ESP con AES (128 bits de clave) en modo CBC - AES-XCBC-MAC-96
  - AH con HMAC-SHA1-96
  - AH con AES-XCBC-MAC-96

En los equipos situados en las redes protegidas por las implementaciones de IPsec se configurarán los mecanismos para establecer conexiones TCP entre dos máquinas, intercambiando información por dichas conexiones según el perfil "*TCP Simétrico*". La velocidad a la que la información se transmitirá y la cantidad de conexiones que se establecerán será variable entre prueba y prueba[8]. En la siguiente lista vemos las combinaciones de ancho de banda por cada túnel criptográfico y cantidad de asociaciones de seguridad que se evaluarán (M es el máximo ancho de banda (en Mbps) para comunicaciones con el perfil "TCP asimétrico", calculado anteriormente):

- $\frac{M}{0{,}5}$ túneles, transmitiendo 0,5 Mbps por cada túnel.
- M túneles, transmitiendo 1 Mbps por cada túnel.
- $\frac{M}{2}$ túneles, transmitiendo 2 Mbps por cada túnel.
- $\frac{M}{3}$ túneles, transmitiendo 3 Mbps por cada túnel.
- $\frac{M}{4}$ túneles, transmitiendo 4 Mbps por cada túnel.
- $\frac{M}{5}$ túneles, transmitiendo 5 Mbps por cada túnel.

[8] El ancho de banda total que debe utilizarse se distribuirá equitativamente entre ambos sentidos. Por lo tanto, una prueba que requiera utilizar un ancho de banda de 2 Mbps por cada túnel enviará 1 Mbps de la implementación de referencia a la implementación evaluada y 1 Mbps en sentido contrario.

En el caso de que se desee evaluar otras velocidades de transmisión, será posible, siempre que la cantidad de asociaciones de seguridad creadas (AS) venga dada por la siguiente expresión (en la que V es la velocidad que se desea medir, en Mbps):

$$AS = \frac{M}{V} \tag{6.1}$$

El uso de compresión en la capa de datos IP no afecta a la medida del número máximo de asociaciones de seguridad simultáneas, aunque se desaconseja para evitar efectos colaterales en la implementación analizada

6.3.6.3. Procedimiento

Para cada combinación de:

- Configuración de Fase 1 de IKE,
- Configuración de Fase 2 de IKE,
- Configuración de ESP / AH, y
- Velocidad de transmisión y asociaciones de seguridad a crear,

se configurarán las implementaciones IPsec de acuerdo a la configuración correspondiente de IKE y ESP / AH. A continuación se comenzarán a establecer las conexiones entre los equipos protegidos por las implementaciones de IPsec, de tal forma que se empezarán a negociar asociaciones de seguridad entre ellas. Al establecerse cada conexión los equipos transmitirán información a la velocidad establecida para la cantidad de asociaciones de seguridad a establecer. El intervalo de tiempo que separará las conexiones nuevas será de 1 segundo, incrementándose en caso de que no sea posible establecer todas las conexiones; es decir, en caso de que:

1. Un intento de establecimiento de conexión resulta fallido.
2. Una conexión previamente establecida se cierre.
3. Una vez establecidas todas las conexiones no sea posible mantenerlas durante al menos 10 segundos.

En caso de fallo se cerrarán todas las conexiones establecidas y se volverá a comenzar, incrementando el intervalo entre cada intento de conexión en un segundo.

Cuando se hayan establecido y mantenido todas las asociaciones de seguridad se dará por concluida la prueba para esta configuración, y se procederá a evaluar la siguiente combinación de factores.

En caso de estimar que el valor inicial de un segundo entre cada establecimiento de conexión es demasiado alto o bajo para la implementación que se está analizando (partiendo, por ejemplo, de la información recogida en la prueba del número máximo de asociaciones de seguridad que la implementación es capaz de establecer simultáneamente), es posible utilizar otro valor que se aproxime más al valor con el que determinaremos el número máximo de asociaciones de seguridad.

6.3.6.4. Mediciones y resultados

Cada vez que una prueba finalice, ya sea satisfactoriamente o no, se almacenará el intervalo de tiempo entre conexiones que se estaba utilizando y el número de asociaciones de seguridad que se ha llegado a establecer. También se almacenará la configuración de IKE y ESP/AH que se estaba utilizando, así como el ancho de banda por cada conexión.

6.3.6.5. Informe de resultados

Los resultados de esta prueba mostrarán el intervalo necesario para el establecimiento de un determinado número de asociaciones de seguridad bajo una configuración concreta, cuando por cada asociación de seguridad se transmite información a una determinada velocidad.

6.3.7. Tiempo de proceso

6.3.7.1. Objetivos

Determinar el tiempo necesario para que la implementación de IPsec que se evalúa lleve a cabo un desarrollo completo de un túnel criptográfico con IPsec, así como la sobrecarga que introduce el cifrado de los datos en la latencia de la red.

6.3.7.2. Configuración necesaria

Las implementaciones de IPsec se configurarán para establecer los túneles criptográficos combinando las siguientes configuraciones de IKE:

- IKE Fase 1
  - Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-MD5-96
  - Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-MD5-96

- Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 2 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 2 (1024 bits), con AES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 2 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96

▪ IKE Fase 2

- Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-MD5-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-MD5-96
- Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 2 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 2 (1024 bits), con AES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y HMAC-SHA1-96
- Grupo de Diffie-Hellman 2 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96
- Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96

Para generar tráfico entre equipos se enviará un mensaje "*Echo Request*" con el campo de datos "`AAAA`" hasta completar un tamaño de 64 bytes.

6.3.7.3. Procedimiento

Para cada configuración diferente para la Fase 1 de IKE se iniciará la negociación de un túnel criptográfico entre una implementación de referencia y la implementación evaluada. La implementación de referencia evaluará el tiempo necesario para completar la negociación de la Fase 1 de IKE y procederá a cancelar la negociación del túnel. Este proceso se repetirá 3 veces para cada configuración de la Fase 1.

Una vez finalizados los estudios de la Fase 1, se llevará a cabo un proceso similar con la Fase 2: Se procederá a negociar una Fase 1 completa con la última configuración probada, para después proceder a la negociación de la Fase 2. Para cada configuración de la Fase 2, la implementación de referencia medirá el tiempo necesario para llevar a cabo la negociación y procederá a cerrar la asociación de seguridad, pasando a la siguiente configuración a evaluar.

Por último, se evaluará el tiempo necesario para el desarrollo completo de la arquitectura de seguridad IPsec. Para ello se cerrarán todas las negociaciones pendientes, y se enviará desde un equipo en la red protegida por la implementación de referencia un mensaje a un equipo protegido por la implementación de IPsec evaluada. El equipo que envía el mensaje medirá el tiempo necesario para obtener el mensaje de respuesta del otro sistema al utilizar las siguientes configuraciones de IPsec:

- IKE (ambas fases): Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-MD5-96; ESP: 3DES en modo CBC y HMAC-MD5-96
- IKE (ambas fases): Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-MD5-96; ESP: 3DES en modo CBC y HMAC-MD5-96
- IKE (ambas fases): Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-SHA1-96; ESP: 3DES en modo CBC y HMAC-SHA1-96
- IKE (ambas fases): Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-SHA1-96; ESP: 3DES en modo CBC y HMAC-SHA1-96
- IKE (ambas fases): Grupo de Diffie-Hellman 2 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96; ESP: AES128 en modo CBC y AES-XCBC-MAC-96
- IKE (ambas fases): Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96; ESP: AES128 en modo CBC y AES-XCBC-MAC-96

- IKE (ambas fases): Grupo de Diffie-Hellman 2 (1024 bits), con 3DES en modo CBC y HMAC-SHA1-96; AH: HMAC-SHA1-96
- IKE (ambas fases): Grupo de Diffie-Hellman 14 (2048 bits), con 3DES en modo CBC y HMAC-SHA1-96; AH: HMAC-SHA1-96
- IKE (ambas fases): Grupo de Diffie-Hellman 2 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96; AH: AES-XCBC-MAC-96
- IKE (ambas fases): Grupo de Diffie-Hellman 14 (2048 bits), con AES en modo CBC y AES-XCBC-MAC-96; AH: AES-XCBC-MAC-96

6.3.7.4. Mediciones y resultados

La implementación de referencia tomará medidas del tiempo necesario desde que comienza la negociación de cada una de las fases de IKE hasta que el proceso de negociación finaliza. Cada una de estas mediciones se llevará a cabo 3 veces, almacenando el valor medio, con el fin de eliminar errores estadísticos y minimizar la influencia de la red en todo este proceso de evaluación.

El equipo que envía el mensaje de Echo Request deberá medir el tiempo que pasa desde que envía el mensaje hasta que recibe la respuesta del otro equipo. Al igual que ocurría anteriormente, este proceso se llevará a cabo 3 veces, recordando siempre destruir las asociaciones de seguridad existentes antes de proceder a la siguiente prueba.

Los resultados de estas pruebas serán los valores medios de los valores capturados en la negociación de cada fase de IKE y en la ejecución completa del protocolo.

6.3.7.5. Informe de resultados

Los resultados de estas pruebas detallarán, para cada suite criptográfica utilizada en la negociación de cada fase, cuáles han sido los valores medidos y cuál es el valor medio de todos ellos. Asimismo, para los tiempos empleados en la ejecución completa del protocolo se mostrará la configuración de IKE y de ESP / AH utilizada, junto con todos los tiempos medidos y el valor medio de dichas medidas.

Capítulo 7

Diseño e Implementación

7.1. Introducción

Al implementar el conjunto de pruebas resultante del trabajo desarrollado aplicando la metodología de validación y evaluación de protocolos de seguridad, se han planteado dos enfoques diferentes pero complementarios. Por un lado se han desarrollado pequeñas pruebas atómicas que desarrollan cada una de las pruebas y evaluaciones descritas en la el capítulo 6. Por otro lado, la implementación de una arquitectura en la que se agrupen todas las pruebas, ofreciendo al usuario ventajas desde el punto de vista de la usabilidad (como un interfaz desde el que configurar las diferentes pruebas a llevar a cabo), representa un paso más hacia la difusión del uso de la metodología aquí propuesta.

En la sección 7.2 se describirá la implementación realizada en forma de pruebas atómicas de cada una de los aspectos en los que se desarrolla el conjunto de pruebas descrito en el capítulo 6. Posteriormente, en la sección 7.3 se estudiarán las características, requisitos y problemática que se presentan para implementar las pruebas en una única plataforma remota desde la que se llevan a cabo tanto los tests como la interpretación de resultados. A partir de este análisis se presentará un diseño de la plataforma, a partir del cuál se llevará a cabo el proceso de desarrollo.

7.2. Pruebas atómicas

Al implementar pequeños programas que realizasen la evaluación de aspectos concretos de una implementación IPsec se plantean múltiples cuestiones, algunas de las cuales ya fueron detectadas al llevar a cabo el análisis de la conformidad con el estándar y del rendimiento (capítulos 3 y 4). En aquellos capítulos se procedió a realizar un estudio de algunos factores que podían originar problemas, tanto desde un punto de vista teórico como al

desarrollar una implementación de las propuestas que allí se hacían.

Por ejemplo, el problema de necesitar una implementación que fuese de antemano conforme con los estándares plantea una complicación eminentemente práctica, ya que no es factible disponer de implementaciones de las que sepamos de antemano que son conformes a estándar, y que podamos controlar hasta el punto de obligar a la implementación a enviar mensajes inválidos durante el establecimiento de asociaciones de seguridad.

Como ya se apuntaba en el propio capítulo 3, la solución elegida ha pasado por adoptar parte de la implementación desarrollada por el NIST americano (llamada Pluto Plus y englobada dentro del proyecto IPsec-WIT ([120]), y combinarla con implementaciones de las que dispusiésemos el código fuente. De esta forma se combinaría la conformidad con los estándares de PlutoPlus, y se le añadirían las nuevas herramientas criptográficas incorporadas a IPsec desde el desarrollo del proyecto IPsec-WIT.

Sin embargo, debido al nivel de modificaciones que ha sido necesario llevar a cabo, la integración esperada ha resultado en unas librerías de programación que permiten comportarse como una implementación de IPsec conforme a los estándares. Estas librerías permiten tanto el establecimiento de túneles criptográficos de acuerdo a las especificaciones de los estándares, como el envío de mensajes de IKE y ESP mal formados[1].

Otras dificultades a las que se ha tenido que hacer frente durante el desarrollo de estas pruebas atómicas han sido los problemas para encontrar implementaciones de las nuevas especificaciones de IPsec (publicadas en Diciembre de 2.005), la necesidad de integrar múltiples herramientas en cada desarrollo atómico y la exigencia de diseñar herramientas propias para solventar problemas que han aparecido en momentos dados.

La dificultad para encontrar implementaciones de las nuevas especificaciones de IPsec ha supuesto un problema de cara al desarrollo de las pruebas. De hecho, en la actualidad únicamente es posible encontrar implementaciones aún en desarrollo y entornos de programación en los que parte de la funcionalidad ya se encuentra disponible, pero el resto queda pendiente para ser programado por nosotros mismos. Esta situación nos ha hecho decantarnos por desarrollar la implementación de las pruebas atómicas únicamente para la especificación de IPsec de 1.998, de la que existen múltiples implementaciones disponibles.

En cuanto a la integración de múltiples herramientas en cada desarrollo atómico, esta situación se daba especialmente en los desarrollos de las pruebas de rendimiento, ya que era necesario integrar generadores de tráfico

[1] En concreto, permite enviar los mensajes correspondientes a la evaluación de la conformidad de los protocolos utilizados para el establecimiento de túneles criptográficos IPsec descritos en el apartado 6.2.4.3 de la aplicación de la metodología a la arquitectura de seguridad IPsec

junto con medidores de tráfico, controladores de tiempo y ancho de banda utilizado y un sistema rudimentario que permitiese informar a todos los equipos de eventos como el inicio del envío de tráfico o el fin de las pruebas a llevar a cabo. Algunos de estos componentes se han podido obtener de herramientas de código abierto (como IPtraf, del que se han obtenido las librerías de medición de tráfico), y otros han sido desarrollados explícitamente para estas pruebas atómicas.

El resultado inmediato de incluir tanta funcionalidad en cada desarrollo atómico ha sido un crecimiento de los recursos necesarios para poder ejecutar las pruebas, así como la redundancia de librerías y funciones comunes en muchos de los desarrollos. De cara a una utilización futura de estos desarrollos, esta sobrecarga de funcionalidad ha originado un código difícil de mantener debido a la variedad de interfaces, estilos de programación que se han integrado en cada una de las pruebas, etc....

Por otro lado, el hecho de tener que desarrollar nuestras propias herramientas en determinados casos ha representado un reto ya que, en la mayoría de ocasiones, ya existían soluciones previas a ese problema. Sin embargo, en muchos casos estas soluciones no se ajustaban a los requisitos del desarrollo que se estaba llevando a cabo, o bien resultaban excesivamente complejas para el objeto que persiguen estas pruebas atómicas. Por ejemplo, en el caso del sistema de notificación entre equipos del inicio y fin de las pruebas, existen múltiples librerías de comunicaciones y control de eventos en red, pero su utilización es demasiado complicada para el fin buscado con estos desarrollos.

Estos dos últimos factores han contribuido a limitar la portabilidad de las pruebas atómicas desarrolladas entre sistemas, poniendo coto a la utilidad de las mismas fuera del entorno en el que han sido implementadas.

Un último problema presente en los desarrollos de pruebas atómicas es la elevada complejidad de su utilización: Al ser herramientas basadas en línea de comandos, estas herramientas no disponen de las ayudas visuales que proporciona un interfaz gráfico al manejar cualquier tipo de software. Además, el hecho de tener que reconfigurar manualmente las implementaciones de IPsec cuando sea necesario para llevar a cabo las pruebas hace que la realización de las mismas no resulte una tarea sencilla.

Sin embargo, y pese a estas dificultades, el desarrollo de las pruebas atómicas ha permitido completar, desde el punto de vista de la implementación, el análisis de los parámetros de conformidad y rendimiento estudiados en los capítulos 3 y 4. Estos desarrollos han permitido comprender mejor qué factores afectan al rendimiento, cómo mejorar la validación de las herramientas criptográficas, o posibles soluciones para solventar los problemas derivados del orden en el que se lleva a cabo la validación de los componentes de IPsec (discutido en el capítulo 3).

Tabla 7.1: Especificaciones de la implementación mediante pruebas atómicas

Sistema Operativo	Linux (kernel 2.6.15)
Lenguaje	C y Bash script
Opciones del Compilador	-Wall -O3
Implementación IPsec utilizada	OpenS/WAN y Pluto Plus
Librerías Criptográficas	OpenSSL 0.9.7j

Adicionalmente, estas pruebas han resultado ser plenamente funcionales, por lo que en la actualidad es posible validar una implementación de IPsec de acuerdo con la especificación del estándar y llevar a cabo una evaluación del rendimiento que dicha implementación puede ofrecer tal y como se describe en el estándar, utilizando estas pruebas atómicas.

Sin embargo, debido a las limitaciones que se comentaban anteriormente en cuanto a portabilidad, y uniendo a esto los problemas relativos a facilidad de uso, su utilización se enfoca más hacia la ayuda y evaluación en el desarrollo de implementaciones de IPsec que hacia el uso por parte de un usuario de dichas implementaciones.

Las características técnicas del entorno en el que se han desarrollado estas pruebas atómicas se resumen en la tabla 7.1.

7.2.1. Relación de pruebas atómicas desarrolladas

A continuación se expondrá la relación completa de las pruebas atómicas desarrolladas, listadas de acuerdo a los diferentes apartados en los que se agrupaban en el capítulo 6 para una mayor facilidad al identificar su funcionalidad.

7.2.1.1. Validación de la conformidad

Validación criptográfica Las pruebas atómicas descritas en las Tablas 7.2, 7.3 y 7.4 han sido desarrolladas con el fin de realizar las pruebas de conformidad de las diferentes herramientas criptográficas utilizadas por las implementaciones de IPsec.

Tabla 7.2: Relación de pruebas atómicas de validación de las herramientas criptográficas utilizadas en ESP

Nombre	Descripción
`EspNull`	Establece un túnel criptográfico utilizando el modo acelerado en la Fase 1 de IKE y el algoritmo NULL en las Fases 1 y 2 de IKE. Protege el tráfico ESP con el algoritmo NULL.
`Esp3Des`	Establece un túnel criptográfico utilizando el modo acelerado en la Fase 1 de IKE y el algoritmo NULL en las Fases 1 y 2 de IKE. Protege el tráfico ESP con el algoritmo 3DES y HMAC-SHA1-96.
`EspAES`	Establece un túnel criptográfico utilizando el modo acelerado en la Fase 1 de IKE y el algoritmo NULL en las Fases 1 y 2 de IKE. Protege el tráfico ESP con el algoritmo AES128 y AES-XCBC-MAC-96.

Tabla 7.3: Relación de pruebas atómicas de validación de las herramientas criptográficas utilizadas en la Fase 1 de IKE

Nombre	Descripción
`Fase13DesMD5`	Establece un túnel criptográfico utilizando el modo acelerado en la Fase 1 de IKE y el algoritmo NULL en la Fase 2 de IKE y en ESP. Protege la Fase 1 de IKE con el algoritmo 3DES y HMAC-MD5-96 y los grupos de Diffie-Hellman 2 y 14.
`Fase13DesSHA1`	Establece un túnel criptográfico utilizando el modo acelerado en la Fase 1 de IKE y el algoritmo NULL en la Fase 2 de IKE y en ESP. Protege la Fase 1 de IKE con el algoritmo 3DES y HMAC-SHA1-96 y los grupos de Diffie-Hellman 2 y 14.
`Fase13DesAES`	Establece un túnel criptográfico utilizando el modo acelerado en la Fase 1 de IKE y el algoritmo NULL en la Fase 2 de IKE y en ESP. Protege la Fase 1 de IKE con el algoritmo 3DES y AES-XCBC-MAC-96 y los grupos de Diffie-Hellman 2 y 14.
`Fase1AesSHA1`	Establece un túnel criptográfico utilizando el modo acelerado en la Fase 1 de IKE y el algoritmo NULL en la Fase 2 de IKE y en ESP. Protege la Fase 1 de IKE con el algoritmo AES128 y HMAC-SHA1-96 y los grupos de Diffie-Hellman 2 y 14.
`Fase1AesAES`	Establece un túnel criptográfico utilizando el modo acelerado en la Fase 1 de IKE y el algoritmo NULL en la Fase 2 de IKE y en ESP. Protege la Fase 1 de IKE con el algoritmo AES128 y AES-XCBC-MAC-96 y los grupos de Diffie-Hellman 2 y 14.

Tabla 7.4: Relación de pruebas atómicas de validación de las herramientas criptográficas utilizadas en la Fase 2 de IKE

Nombre	Descripción
`Fase23DesMD5`	Establece un túnel criptográfico utilizando el modo acelerado y el algoritmo NULL en la Fase 1 de IKE y en ESP. Protege la Fase 2 de IKE con el algoritmo 3DES y HMAC-MD5-96 y los grupos de Diffie-Hellman 2 y 14.
`Fase23DesSHA1`	Establece un túnel criptográfico utilizando el modo acelerado y el algoritmo NULL en la Fase 1 de IKE y en ESP. Protege la Fase 2 de IKE con el algoritmo 3DES y HMAC-SHA1-96 y los grupos de Diffie-Hellman 2 y 14.
`Fase23DesAES`	Establece un túnel criptográfico utilizando el modo acelerado y el algoritmo NULL en la Fase 1 de IKE y en ESP. Protege la Fase 2 de IKE con el algoritmo 3DES y AES-XCBC-MAC-96 y los grupos de Diffie-Hellman 2 y 14.
`Fase2AesSHA1`	Establece un túnel criptográfico utilizando el modo acelerado y el algoritmo NULL en la Fase 1 de IKE y en ESP. Protege la Fase 2 de IKE con el algoritmo AES128 y HMAC-SHA1-96 y los grupos de Diffie-Hellman 2 y 14.
`Fase2AesAES`	Establece un túnel criptográfico utilizando el modo acelerado y el algoritmo NULL en la Fase 1 de IKE y en ESP. Protege la Fase 2 de IKE con el algoritmo AES128 y AES-XCBC-MAC-96 y los grupos de Diffie-Hellman 2 y 14.

Validación de protocolos Las pruebas atómicas que se pueden observar en las Tablas 7.5 y 7.6 tienen por objeto el análisis del desarrollo de los diferentes protocolos de IPsec que lleva a cabo una implementación de esta arquitectura de seguridad. Todas estas pruebas establecen un túnel criptográfico de acuerdo a los parámetros indicados en la descripción de cada una, y posteriormente lleva a cabo el envío de mensajes especialmente formados descritos en la sección 6.2.4.

Tabla 7.5: Relación de pruebas atómicas de validación del desarrollo de los protocolos en modo túnel

Nombre	Descripción
TunelMMQMESP	Establece un túnel criptográfico en modo túnel utilizando Main Mode para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y ESP para proteger el tráfico. No utiliza Perfect Forward Secrecy.
TunelMMQMAH	Establece un túnel criptográfico en modo túnel utilizando Main Mode para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y AH para proteger el tráfico. No utiliza Perfect Forward Secrecy.
TunelMMQMESPPFS	Establece un túnel criptográfico en modo túnel utilizando Main Mode para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y ESP para proteger el tráfico. Utiliza Perfect Forward Secrecy.
TunelMMQMAHPFS	Establece un túnel criptográfico en modo túnel utilizando Main Mode para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y AH para proteger el tráfico. Utiliza Perfect Forward Secrecy.
TunelAMQMESP	Establece un túnel criptográfico en modo túnel utilizando Modo Acelerado para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y ESP para proteger el tráfico. No utiliza Perfect Forward Secrecy.
TunelAMQMAH	Establece un túnel criptográfico en modo túnel utilizando Modo Acelerado para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y AH para proteger el tráfico. No utiliza Perfect Forward Secrecy.
TunelAMQMESPPFS	Establece un túnel criptográfico en modo túnel utilizando Modo Acelerado para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y ESP para proteger el tráfico. Utiliza Perfect Forward Secrecy.
TunelAMQMAHPFS	Establece un túnel criptográfico en modo túnel utilizando Modo Acelerado para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y AH para proteger el tráfico. Utiliza Perfect Forward Secrecy.

Tabla 7.6: Relación de pruebas atómicas de validación del desarrollo de los protocolos en modo transporte

Nombre	Descripción
TranspMMQMESP	Establece un túnel criptográfico en modo transporte utilizando Main Mode para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y ESP para proteger el tráfico. No utiliza Perfect Forward Secrecy.
TranspMMQMAH	Establece un túnel criptográfico en modo transporte utilizando Main Mode para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y AH para proteger el tráfico. No utiliza Perfect Forward Secrecy.
TranspMMQMESPPFS	Establece un túnel criptográfico en modo transporte utilizando Main Mode para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y ESP para proteger el tráfico. Utiliza Perfect Forward Secrecy.
TranspMMQMAHPFS	Establece un túnel criptográfico en modo transporte utilizando Main Mode para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y AH para proteger el tráfico. Utiliza Perfect Forward Secrecy.
TranspAMQMESP	Establece un túnel criptográfico en modo transporte utilizando Modo Acelerado para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y ESP para proteger el tráfico. No utiliza Perfect Forward Secrecy.
TranspAMQMAH	Establece un túnel criptográfico en modo transporte utilizando Modo Acelerado para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y AH para proteger el tráfico. No utiliza Perfect Forward Secrecy.
TranspAMQMESPPFS	Establece un túnel criptográfico en modo transporte utilizando Modo Acelerado para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y ESP para proteger el tráfico. Utiliza Perfect Forward Secrecy.
TranspAMQMAHPFS	Establece un túnel criptográfico en modo transporte utilizando Modo Acelerado para la Fase 1 de IKE, Quick Mode para la Fase 2 de IKE y AH para proteger el tráfico. Utiliza Perfect Forward Secrecy.

Tabla 7.7: Relación de pruebas atómicas de validación de los mecanismos de autenticación

Nombre	Descripción
PskMd5MM	Lleva a cabo una autenticación utilizando un secreto compartido, utilizando MD5 como función resumen. En la Fase 1 de IKE se utiliza Main Mode.
PskMd5AM	Lleva a cabo una autenticación utilizando un secreto compartido, utilizando MD5 como función resumen. En la Fase 1 de IKE se utiliza Modo Acelerado.
PskSha1MM	Lleva a cabo una autenticación utilizando un secreto compartido, utilizando SHA1 como función resumen. En la Fase 1 de IKE se utiliza Main Mode.
PskSha1AM	Lleva a cabo una autenticación utilizando un secreto compartido, utilizando SHA1 como función resumen. En la Fase 1 de IKE se utiliza Modo Acelerado.
RsaMM	Lleva a cabo una autenticación utilizando pares de claves RSA. En la Fase 1 de IKE se utiliza Main Mode.
RsaAM	Lleva a cabo una autenticación utilizando pares de claves RSA. En la Fase 1 de IKE se utiliza Modo Acelerado.
X509MM	Lleva a cabo una autenticación utilizando certificados X.509. En la Fase 1 de IKE se utiliza Main Mode.
X509AM	Lleva a cabo una autenticación utilizando certificados X.509. En la Fase 1 de IKE se utiliza Modo Acelerado.

Validación de los mecanismos de autenticación Con el fin de validar el desarrollo de los diferentes mecanismos de autenticación presentes en una implementación de IPsec contamos con las pruebas atómicas enumeradas en la Tabla 7.7. Cada una de estas implementaciones lleva a cabo las tres pruebas especificadas, esto es: autenticación correcta, envío de credenciales incorrectas y recepción de credenciales incorrectas.

Tabla 7.8: Relación de pruebas atómicas de validación de la gestión de claves

Nombre	Descripción
`Fase1Tiempo`	Establece un túnel criptográfico mediante IPsec en el que se negocia la caducidad de las claves de la Fase 1 de IKE en función del tiempo. El valor por defecto es de 30 segundos, aunque es modificable mediante parámetros. La Fase 2 tiene un tiempo de caducidad de 1 hora.
`Fase2Tiempo`	Establece un túnel criptográfico mediante IPsec en el que se negocia la caducidad de las claves de la Fase 2 de IKE en función del tiempo. El valor por defecto es de 30 segundos, aunque es modificable mediante parámetros. La Fase 1 tiene un tiempo de caducidad de 1 hora.
`Fase1Trafico`	Establece un túnel criptográfico mediante IPsec en el que se negocia la caducidad de las claves de la Fase 1 de IKE en función del volumen de tráfico protegido. El valor por defecto es de 10 KBytes, aunque es modificable mediante parámetros. La Fase 2 tiene un tiempo de caducidad de 1 hora.
`Fase2Trafico`	Establece un túnel criptográfico mediante IPsec en el que se negocia la caducidad de las claves de la Fase 2 de IKE en función del volumen de tráfico protegido. El valor por defecto es de 10 KBytes, aunque es modificable mediante parámetros. La Fase 1 tiene un tiempo de caducidad de 1 hora.

Validación de los mecanismos de gestión de claves En la Tabla 7.8 podemos ver un listado completo de las pruebas atómicas desarrolladas con el fin de evaluar el grado de conformidad de los mecanismos de renovación y caducidad de las claves con respecto a lo especificado en el estándar.

Tabla 7.9: Relación de pruebas atómicas de validación de otras características adicionales

Nombre	Descripción
`NAT`	Establece, secuencialmente, tres túneles criptográficos, de acuerdo a los requisitos especificados en el conjunto de pruebas. Entre el establecimiento de cada túnel se solicita al usuario que revise y actualice en caso necesario la configuración de la implementación evaluada.
`IPComp`	Establece un túnel criptográfico en el que se utiliza IPComp, transmitiendo por dicho túnel mensajes que deben generar una respuesta determinada.
`ECN`	Establece un túnel criptográfico y posteriormente satura la red, obligando a las implementaciones a notificar de la existencia de congestión en la red.

Validación de otras características Para la implementación de los últimos análisis de conformidad necesarios, se han desarrollado las pruebas atómicas que aparecen en la Tabla 7.9.

7.2.1.2. Evaluación del rendimiento

Ancho de banda Las pruebas atómicas que se detallan en la Tabla 7.10 han sido desarrolladas con el fin de llevar a cabo una medición fiable del ancho que banda que una implementación de IPsec puede ofrecer. Estas pruebas atómicas **no establecen túneles criptográficos**, sino que utilizan la implementación de referencia para ello. Por lo tanto, cualquier variación respecto a los parámetros del túnel se lleva a cabo en la configuración de la implementación de referencia, y no en las pruebas atómicas. Es necesario destacar que debido al carácter compacto de las pruebas atómicas, así como a la complejidad de la inclusión de la funcionalidad necesaria, no se han implementado las pruebas atómicas de evaluación del rendimiento que corresponden a perfiles de tráfico de redes saturadas.

Tabla 7.10: Relación de pruebas atómicas de evaluación del ancho de banda.

Nombre	Descripción
`BWTMax-Serv`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "Máximo Ancho de Banda". Este módulo lleva a cabo las funciones de servidor, por lo que espera las conexiones entrantes. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.
`BWTMax-Clie`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "Máximo Ancho de Banda". Este módulo lleva a cabo las funciones de cliente, por lo que se conecta al servidor instalado en otro dispositivo y comienza el envío de tráfico. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.
`Trad-Serv`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "Usuario Tradicional". Este módulo lleva a cabo las funciones de servidor, por lo que espera las conexiones entrantes. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.
`Trad-Clie`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "Usuario Tradicional". Este módulo lleva a cabo las funciones de cliente, por lo que se conecta al servidor instalado en otro dispositivo y comienza el envío de tráfico. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.

Nombre	Descripción
`TCPAsimIn-Serv`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "TCP Asimétrico". Este módulo lleva a cabo las funciones de servidor, por lo que espera las conexiones entrantes y comienza el envío de tráfico. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.
`TCPAsimIn-Clie`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "TCP Asimétrico Entrante". Este módulo lleva a cabo las funciones de cliente, por lo que se conecta al servidor instalado en otro dispositivo y comienza el envío de tráfico. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.
`TCPAsimOut-Serv`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "TCP Asimétrico Saliente". Este módulo lleva a cabo las funciones de servidor, por lo que espera las conexiones entrantes y procede al envío de tráfico. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.
`TCPAsimOut-Clie`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "TCP Asimétrico Saliente". Este módulo lleva a cabo las funciones de cliente, por lo que se conecta al servidor instalado en otro dispositivo y no envía tráfico alguno (salvo los ACKs de TCP que se envían automáticamente). Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.

Nombre	Descripción
`TCPSim-Serv`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "Usuario Tradicional". Este módulo lleva a cabo las funciones de servidor, por lo que espera las conexiones entrantes. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.
`TCPSim-Clie`	Genera tráfico entre dos equipos separados por un túnel criptográfico de acuerdo a los parámetros del perfil "Usuario Tradicional". Este módulo lleva a cabo las funciones de cliente, por lo que se conecta al servidor instalado en otro dispositivo y comienza el envío de tráfico. Tras 10 segundos de transmisión mide el ancho de banda utilizado de media a intervalos de 1 segundo, durante 5 segundos. Posteriormente cierra la conexión y repite el proceso completo otras 2 veces. Finalmente, informa del ancho de banda medio que se ha estado transmitiendo durante las medidas.

Máximo número de asociaciones de seguridad simultáneas Para llevar a cabo la evaluación del máximo número de asociaciones de seguridad simultáneas que una implementación de IPsec puede establecer, se han desarrollado las pruebas atómicas que aparecen descritas en la Tabla 7.11. Estas pruebas atómicas **no establecen túneles criptográficos**, sino que utilizan la implementación de referencia para ello. Su labor se centra, por lo tanto, en la evaluación del rendimiento.

Establecimiento de nuevas asociaciones de seguridad En la Tabla 7.12 se pueden consultar cuáles han sido las pruebas atómicas desarrolladas para permitir la evaluación de la capacidad de establecimiento de nuevas asociaciones de seguridad de la implementación de IPsec evaluada. Estas pruebas atómicas **no establecen túneles criptográficos**, sino que utilizan la implementación de referencia para ello. Su labor se centra, por lo tanto, en la evaluación del rendimiento.

Tiempo de proceso Por último, las pruebas atómicas disponibles para permitirnos medir el tiempo de proceso necesario para el desarrollo completo

Tabla 7.11: Relación de pruebas atómicas de evaluación del número máximo de asociaciones de seguridad

Nombre	Descripción
`MaxSA-Servidor`	Establece servidores en el equipo en el que se instala a medida que se establecen conexiones, de forma que se puedan recibir comunicaciones entrantes. Cuando alguna conexión falla evalúa si es necesario repetir la medición (porque únicamente hay una medida o porque los resultados de las dos últimas pruebas no coinciden), y en caso necesario vuelve a comenzar el proceso.
`MaxSA-Cliente`	Establece conexiones con el equipo en el que se ejecuta el módulo servidor a intervalos regulares. Cuando alguna conexión falla evalúa si es necesario repetir la medición (porque únicamente hay una medida o porque los resultados de las dos últimas pruebas no coinciden), y en caso necesario vuelve a comenzar el proceso.

de los diferentes protocolos de IPsec aparecen en las Tablas 7.13, 7.14 y 7.15.

Tabla 7.12: Relación de pruebas atómicas de evaluación de la capacidad de establecimiento de nuevas asociaciones de seguridad

Nombre	Descripción
`CapacidadSA-Servidor`	Establece servidores en el equipo en el que se instala a medida que se establecen conexiones, de forma que se puedan recibir comunicaciones entrantes. Cuando alguna conexión falla evalúa si es necesario repetir la medición (si no se han establecido el número de túneles especificado), y en caso necesario vuelve a comenzar el proceso.
`CapacidadSA-Cliente`	Establece conexiones con el equipo en el que se ejecuta el módulo servidor a intervalos regulares, enviando datos por cada túnel de tal forma que el ancho de banda total se reparta equitativamente entre todos los túneles. Cuando alguna conexión falla ajusta el periodo de espera entre el establecimiento de los túneles y vuelve a comenzar el proceso.

Tabla 7.13: Relación de pruebas atómicas de evaluación del tiempo de proceso de la Fase 1 de IKE

Nombre	Descripción
`TiempoF1-3desMd5`	Establece un túnel criptográfico utilizando en la Fase 1 los algoritmos 3DES y HMAC-MD5-96. Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.
`TiempoF1-3desSha1`	Establece un túnel criptográfico utilizando en la Fase 1 los algoritmos 3DES y HMAC-SHA1-96. Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.
`TiempoF1-3desAes`	Establece un túnel criptográfico utilizando en la Fase 1 los algoritmos 3DES y AES-XCBC-MAC-96.Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.
`TiempoF1-AesSha1`	Establece un túnel criptográfico utilizando en la Fase 1 los algoritmos AES128 y HMAC-SHA1-96.Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.
`TiempoF1-AesAes`	Establece un túnel criptográfico utilizando en la Fase 1 los algoritmos AES128 y AES-XCBC-MAC-96.Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.

Tabla 7.14: Relación de pruebas atómicas de evaluación del tiempo de proceso de la Fase 2 de IKE

Nombre	Descripción
`TiempoF2-3desMd5`	Establece un túnel criptográfico utilizando en la Fase 2 los algoritmos 3DES y HMAC-MD5-96. Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.
`TiempoF2-3desSha1`	Establece un túnel criptográfico utilizando en la Fase 2 los algoritmos 3DES y HMAC-SHA1-96. Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.
`TiempoF2-3desAes`	Establece un túnel criptográfico utilizando en la Fase 2 los algoritmos 3DES y AES-XCBC-MAC-96.Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.
`TiempoF2-AesSha1`	Establece un túnel criptográfico utilizando en la Fase 2 los algoritmos AES128 y HMAC-SHA1-96.Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.
`TiempoF2-AesAes`	Establece un túnel criptográfico utilizando en la Fase 2 los algoritmos AES128 y AES-XCBC-MAC-96.Negocia varias configuraciones diferentes, haciendo uso de los grupos de Diffie-Hellman 2 y 14, y habilitando o deshabilitando PFS. Realiza 3 medidas de cada configuración.

Tabla 7.15: Relación de pruebas atómicas de evaluación del tiempo de establecimiento de un túnel criptográfico completo

Nombre	Descripción
T-3desMd5-ESP-3desMd5	Establece un túnel criptográfico utilizando en IKE los algoritmos 3DES y HMAC-MD5-96. Protege el tráfico utilizando ESP con los algoritmos 3DES y HMAC-MD5-96. Utiliza cuatro configuraciones diferentes, habilitando y deshabilitando el uso de PFS, y utilizando los grupos de Diffie-Hellman 2 y 14. Realiza 3 medidas de cada configuración.
T-3desSha1-ESP-3desSha1	Establece un túnel criptográfico utilizando en IKE los algoritmos 3DES y HMAC-SHA1-96. Protege el tráfico utilizando ESP con los algoritmos 3DES y HMAC-SHA1-96. Utiliza cuatro configuraciones diferentes, habilitando y deshabilitando el uso de PFS, y utilizando los grupos de Diffie-Hellman 2 y 14. Realiza 3 medidas de cada configuración.
T-AesAes-ESP-AesAes	Establece un túnel criptográfico utilizando en IKE los algoritmos AES128 y AES-XCBC-HMAC-96. Protege el tráfico utilizando ESP con los algoritmos 3DES y HMAC-MD5-96. Utiliza cuatro configuraciones diferentes, habilitando y deshabilitando el uso de PFS, y utilizando los grupos de Diffie-Hellman 2 y 14. Realiza 3 medidas de cada configuración.
T-3desSha1-AH-Sha1	Establece un túnel criptográfico utilizando en IKE los algoritmos 3DES y HMAC-SHA1-96. Protege el tráfico utilizando AH con el algoritmo HMAC-SHA1-96. Utiliza cuatro configuraciones diferentes, habilitando y deshabilitando el uso de PFS, y utilizando los grupos de Diffie-Hellman 2 y 14. Realiza 3 medidas de cada configuración.
T-AesAes-AH-Aes	Establece un túnel criptográfico utilizando en IKE los algoritmos AES128 y AES-XCBC-HMAC-96. Protege el tráfico utilizando AH con el algoritmo AES-XCBC-HMAC-96. Utiliza cuatro configuraciones diferentes, habilitando y deshabilitando el uso de PFS, y utilizando los grupos de Diffie-Hellman 2 y 14. Realiza 3 medidas de cada configuración.

7.2.2. Ejemplos de ejecución de pruebas atómicas

En este apartado se mostrarán dos ejemplos de ejecución de las pruebas atómicas, de modo que sirvan como ejemplo de la estructura y funcionamiento del conjunto completo de pruebas detallado en el apartado anterior. Además del resultado de la ejecución de las pruebas se incluye también un diagrama explicativo acerca de las operaciones internas de cada una de las pruebas atómicas

7.2.2.1. Ejemplo 1: Autenticación mediante secreto compartido (PSK) en Main Mode de IKE

Para validar la autenticación mediante secreto compartido en IKE al utilizar Main Mode se ha desarrollado una aplicación que negocia el establecimiento de túneles IPsec hasta el final de la Fase 1 de IKE, autenticándose mediante secreto compartido. Esta aplicación lleva a cabo, para cada función resumen seleccionada, tres pruebas diferentes:

- Autenticación correcta (ambas implementaciones envían credenciales correctas)
- Envío de credenciales incorrectas
- Recepción de credenciales incorrectas

Para poder llevar a cabo las tres pruebas sin requerir al usuario que altere la contraseña en la implementación evaluada, la aplicación automáticamente lleva a cabo modificaciones en el resumen de la contraseña esperada, de forma que al recibir las credenciales de la implementación evaluada, éstas serán rechazadas. El grafo de ejecución de esta prueba atómica puede verse en la Figura 7.1.

Un ejemplo de la ejecución de esta prueba atómica puede verse en la Tabla 7.16. Aquí se puede ver cómo la aplicación necesita recibir como parámetros las funciones resumen a utilizar, la dirección de red en la que se encuentra la implementación de IPsec a evaluar, y el secreto compartido que se debe utilizar[2].

[2]A la aplicación se le debe suministrar el secreto compartido correcto, ya que ella se encarga de alterarla en los casos en los que sea necesario

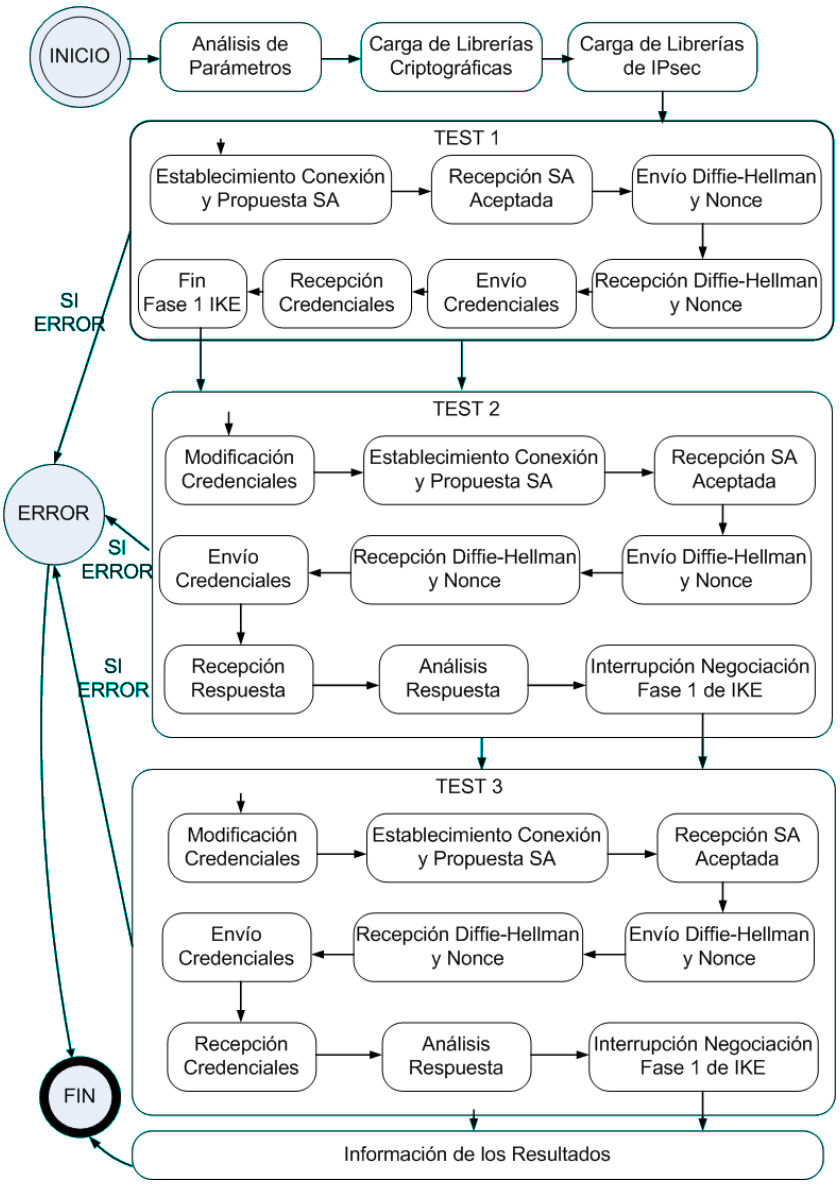

Figura 7.1: Diagrama de estados de la prueba atómica de validación de la autenticación utilizando secreto compartido en Main Mode de IKE

Tabla 7.16: Validación de la autenticación mediante secreto compartido

```
usuario@host# ./validacion/auth-psk-mm --resumen md5
--direccion 192.168.0.100 --clave "abcdefghijklmnop"

Test 1: Autenticacion Correcta
Estableciendo Conexion ... Establecida
Enviando Propuesta de SA ... Enviada
Recibiendo SA aceptada ... Recibida y OK
Enviando Diffie-Hellman y Nonce ... Enviados
Recibiendo Diffie-Hellman y Nonce ... Recibido y OK
Enviando Credenciales ... Enviadas
Recibiendo Credenciales ... Recibidas y OK
Fase 1 negociada con exito. Fin del test 1


Test 2: Envio de Credenciales Invalidas
Modificando credenciales ... Hecho
Estableciendo Conexion ... Establecida
Enviando Propuesta de SA ... Enviada
Recibiendo SA aceptada ... Recibida y OK
Enviando Diffie-Hellman y Nonce ... Enviados
Recibiendo Diffie-Hellman y Nonce ... Recibido y OK
Enviando Credenciales ... Enviadas
Recibiendo Credenciales ... Recibidas -- Rechazo
Fase 1 interrumpida. Fin del test 2


Test 3: Recepcion de Credenciales Invalidas
Modificando credenciales ... Hecho
Estableciendo Conexion ... Establecida
Enviando Propuesta de SA ... Enviada
Recibiendo SA aceptada ... Recibida y OK
Enviando Diffie-Hellman y Nonce ... Enviados
Recibiendo Diffie-Hellman y Nonce ... Recibido y OK
Enviando Credenciales ... Enviadas
Recibiendo Credenciales ... Recibidas y OK
Rechazando Credenciales ... Hecho
Fase 1 interrumpida. Fin del test 3
```

```
RESULTADOS
Test 1:
 Obtenido: FASE 1 OK
 Esperado: FASE 1 OK
Test 2:
 Obtenido: Nuestras Credenciales Rechazadas
 Esperado: Nuestras Credenciales Rechazadas
Test 3:
 Obtenido: Rechazamos Credenciales
 Esperado: Rechazamos Credenciales
RESULTADO FINAL: 3/3
Autenticacion mediante PSK en MM con MD5 conforme

usuario@host#
```

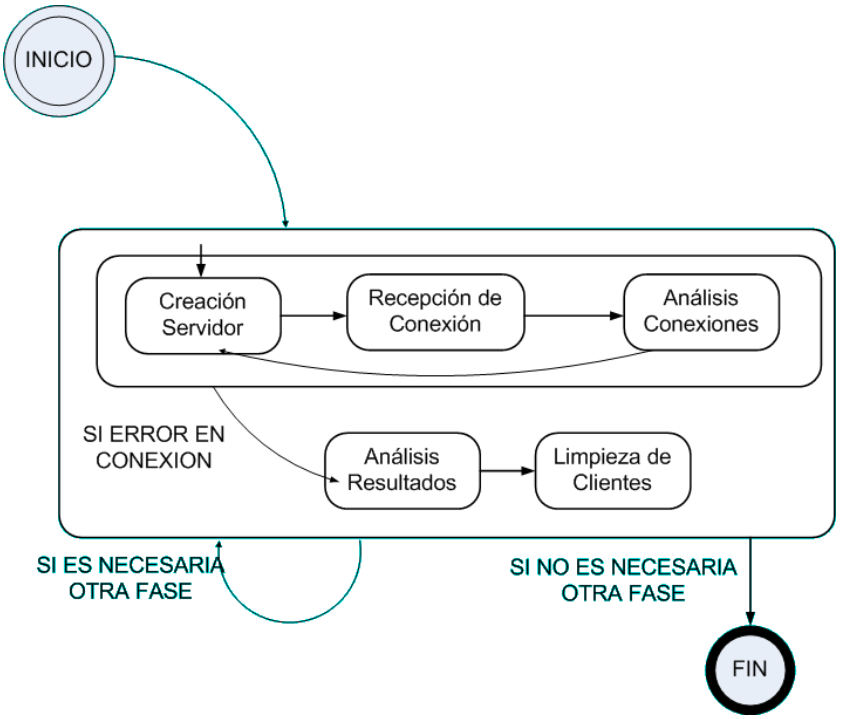

Figura 7.2: Diagrama de estados del módulo servidor de la prueba de evaluación del número máximo de SA que puede establecer una implementación de IPsec

7.2.2.2. Ejemplo 2: Máximo número de asociaciones de seguridad

La aplicación que desarrolla las pruebas atómicas para la evaluación de la capacidad de establecimiento de asociaciones de seguridad de una implementación de IPsec consta de dos módulos independientes. El primero de ellos (módulo servidor) se encarga de crear servidores TCP que se encarguen de aceptar las conexiones entrantes, conexiones que generarán el establecimiento de nuevos túneles criptográficos. Este módulo se ejecutará en la red protegida por la implementación evaluada.

Por su parte, el módulo cliente se encargará de crear clientes que se comunicarán con los servidores creados por el módulo servidor. Además, este módulo será el encargado de controlar la velocidad a la que se establecen las conexiones, así como el éxito o no de dicha conexión.

Los diagramas de estado de esta prueba se pueden ver en las Figuras 7.2 (módulo servidor) y 7.2.2.2 (módulo cliente).

Por su parte, el resultado de la ejecución del módulo servidor puede verse en la Tabla 7.17. En ella podemos ver cómo el módulo servidor no requiere de datos adicionales por parte del usuario. Sin embargo, en la Tabla

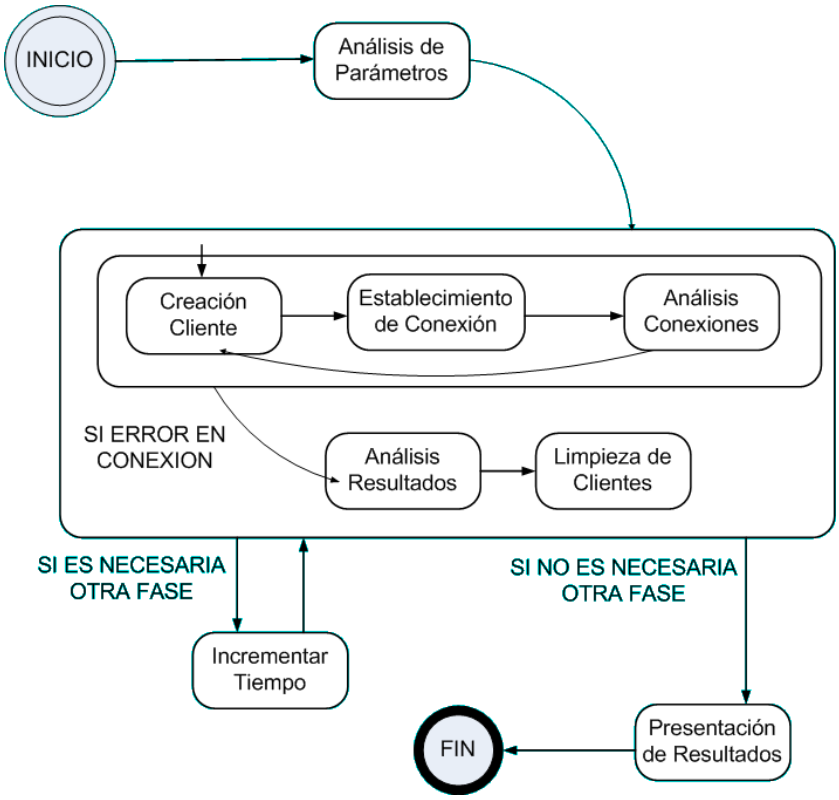

Figura 7.3: Diagrama de estados del módulo cliente de la prueba de evaluación del número máximo de SA que puede establecer una implementación de IPsec

7.18 vemos cómo, en la ejecución del cliente, es necesario incluir parámetros adicionales, que se corresponden con el tiempo inicial que debe pasar entre cada inicio de conexión y la dirección de red del equipo en el que se ejecuta el módulo servidor.

Tabla 7.17: Ejecución del módulo servidor de la evaluación de cantidad de asociaciones de seguridad simultáneas que soporta la implementación de IPsec

```
usuario@host# ./evaluacion/maxsa-servidor
Iniciando Fase 1
Iniciando servidor #1...
Conexion establecida... Iniciado servidor #2
Conexion establecida... Iniciado servidor #3
[...]
Conexion establecida... Iniciado servidor #254
Conexion establecida... Iniciado servidor #255
Conexion establecida... Iniciado servidor #256

ALERTA: Se pierde la conexion #1
Cerrando todos los servidores...

Iniciando Fase 2
Iniciando servidor #1...
Conexion establecida... Iniciado servidor #2
Conexion establecida... Iniciado servidor #3
[...]
Conexion establecida... Iniciado servidor #254
Conexion establecida... Iniciado servidor #255
Conexion establecida... Iniciado servidor #256
Conexion establecida... Iniciado servidor #257
[...]
Conexion establecida... Iniciado servidor #264
Conexion establecida... Iniciado servidor #265

ALERTA: Se pierde la conexion #76
Cerrando todos los servidores...
Es necesario utilizar otra Fase
```

```
Iniciando Fase 3
Iniciando servidor #1...
Conexion establecida... Iniciado servidor #2
Conexion establecida... Iniciado servidor #3
[...]
Conexion establecida... Iniciado servidor #254
Conexion establecida... Iniciado servidor #255
Conexion establecida... Iniciado servidor #256
Conexion establecida... Iniciado servidor #257
[...]
Conexion establecida... Iniciado servidor #264
Conexion establecida... Iniciado servidor #265

ALERTA: Se pierde la conexion #152
Cerrando todos los servidores...
Prueba Finalizada

usuario@host#
```

Tabla 7.18: Ejecución del módulo cliente de la evaluación de cantidad de asociaciones de seguridad simultáneas que soporta la implementación de IPsec

```
usuario@host# ./evaluacion/maxsa-cliente 1
192.168.100.10
Iniciando Fase 1 (tiempo entre conexiones, 1 segundo)
Estableciendo conexion #1... conexion establecida
Estableciendo conexion #2... conexion establecida
Estableciendo conexion #3... conexion establecida
[...]
Estableciendo conexion #254... Conexion establecida
Estableciendo conexion #255... Conexion establecida
Estableciendo conexion #256... Conexion NO establecida

ALERTA: conexion #256 no establecida (TimeOut)
Cerrando todas las conexiones...

Iniciando Fase 2 (tiempo entre conexiones, 2 segundos)
Estableciendo conexion #1... conexion establecida
Estableciendo conexion #2... conexion establecida
Estableciendo conexion #3... conexion establecida
[...]
Estableciendo conexion #254... Conexion establecida
Estableciendo conexion #255... Conexion establecida
Estableciendo conexion #256... Conexion establecida
Estableciendo conexion #257... Conexion establecida
[...]
Estableciendo conexion #264... Conexion establecida

ALERTA: Se pierde la conexion #76
Cerrando todas las conexiones...
Es necesario utilizar otra Fase
```

```
Iniciando Fase 3 (tiempo entre conexiones, 3 segundos)
Estableciendo conexion #1... conexion establecida
Estableciendo conexion #2... conexion establecida
Estableciendo conexion #3... conexion establecida
[...]
Estableciendo conexion #254... Conexion establecida
Estableciendo conexion #255... Conexion establecida
Estableciendo conexion #256... Conexion establecida
Estableciendo conexion #257... Conexion establecida
[...]
Estableciendo conexion #264... Conexion establecida

ALERTA: Se pierde la conexion #152
Cerrando todas las conexiones...
Prueba Finalizada

La implementacion es capaz de establecer,
simultaneamente, 263 asociaciones de seguridad

usuario@host#
```

Como podemos ver, en la ejecución de esta prueba se han llevado a cabo 3 fases en las que se establecían tantas asociaciones de seguridad como resultaba posible. En la Fase 1 de dicha ejecución una escasez de recursos en la implementación analizada ha originado que el establecimiento de una asociación de seguridad nueva no se pudiese llevar a cabo. Sin embargo, ampliando el tiempo entre cada establecimiento de conexión ha sido posible establecer un número mayor de asociaciones de seguridad.

7.3. Diseño de la plataforma de validación y evaluación remota

De cara a proporcionar una plataforma desde la que realizar la validación y evaluación de las implementaciones IPsec de forma conveniente y con mayor facilidad de uso que la que se lleva a cabo con las pruebas atómicas, la implementación de una plataforma que desarrollase el conjunto de pruebas generado a partir de la metodología propuesta en esta tesis de forma remota mediante un interfaz amigable es el siguiente paso a llevar a cabo.

Los requisitos que se plantean de cara al desarrollo de esta arquitectura son los siguientes:

- **Validación y Evaluación Remotas**: La plataforma deberá ser capaz de llevar a cabo el desarrollo de todas las pruebas remotamente, es decir, sin que el usuario de la plataforma deba estar conectado al mismo segmento de red en el que se encuentra la plataforma. Idealmente cualquier usuario debería poder indicar la dirección de la implementación de IPsec que desea evaluar y las pruebas se empezarían a procesar automáticamente, sin mayor intervención del usuario. Sin embargo, aspectos tales como la necesidad de disponer de software generador y medidor del tráfico de red en la red protegida por la implementación de IPsec que se analiza, la necesidad de reconfigurar la implementación de IPsec múltiples veces a lo largo de las pruebas, etc... hacen que la intervención del usuario durante el desarrollo de las pruebas sea inevitable.

  Aún así, al implementar esta arquitectura se estudiarán las implementaciones atómicas de las que ya se dispone, para optimizarlas en cuanto a usabilidad e intervención del usuario requerida, de tal forma que la intervención del usuario pueda ser reducida al mínimo posible.

  En cuanto al hecho de que el usuario utilice la red para conectarse a la plataforma, este hecho introduce una serie de retos y restricciones que es interesante considerar. Por un lado, al habilitar el acceso mediante las tecnologías de red a un interfaz en el que se configuran las pruebas a llevar a cabo y se obtienen los resultados, es importante

diseñar el interfaz de tal forma que la amplia variedad de dispositivos que hoy día pueden hacer uso de las redes de comunicaciones y que pueden implementar la arquitectura de seguridad IPsec sean capaces de hacer uso del interfaz de forma sencilla, eficaz y consistente entre plataformas. Este problema se solucionará utilizando tecnologías estandarizadas para los contenidos en Internet, principalmente aquellas incluidas en la tecnología AJAX ([59]), ya que todas ellas se encuentran estandarizadas por el World Wide Web Consortium (W3C) y su integración permite el manejo dinámico de la información que proporcione un origen de datos asíncrono.

Por otro lado, el hecho de que el usuario pueda no encontrarse en la misma red que la implementación que se desea analizar hace que sea necesario desarrollar nuevas herramientas que permitan obtener información que normalmente se obtendría del usuario (como por ejemplo, el valor del MTU, necesario para las pruebas de rendimiento). Estas herramientas deberán integrarse en el resto de la plataforma de forma que, en caso necesario, podamos obtener esos datos sin intervención del usuario.

Adicionalmente es necesario tener en cuenta que el hecho de utilizar una plataforma basada en la red para llevar a cabo evaluaciones del rendimiento de la red puede introducir conflictos y problemas que devalúen la información proporcionada. Por ejemplo, al evaluar el máximo ancho de banda que una implementación de IPsec puede proporcionar es importante que la infraestructura de red entre la implementación que genera las pruebas y la implementación evaluada cuente con capacidad para soportar todo el tráfico necesario para saturar la implementación evaluada. Sin embargo, las tecnologías de conexión a redes remotas (bien sea mediante acceso telefónico, ADSL, etc...) cuentan con un ancho de banda máximo menor que cualquiera de las tecnologías de red de área local utilizadas en la actualidad (siendo éstas las que se encuentran soportadas por las implementaciones IPsec), por lo que los resultados obtenidos pueden verse desvirtuados.

Estos problemas deberán ser estudiados y analizados antes de decidir un diseño definitivo para la plataforma, con el fin de evitar utilizar diseños que posteriormente representen un problema para solucionar estas cuestiones.

- **Modularidad**: La implementación deberá ser modular, en el sentido de que la inclusión de nuevos conjuntos de pruebas deberá ser una tarea viable. De esta forma estaremos previniendo la obsolescencia de la plataforma en cuanto se lleve a cabo alguna modificación de los estándares, al tiempo que facilitamos la ampliación de las pruebas que se llevan a cabo. Por otro lado la actualización de la plataforma en

caso necesario se convertirá en una tarea más asequible al necesitar actualizar únicamente aquellos módulos que lo requieren, en lugar de toda la plataforma.

Para afrontar este requisito deberá diseñarse unas interfaces de programación y ficheros de descripción de pruebas a realizar desde los que se accederá a los desarrollos de las pruebas en sí, permitiendo de esta forma que cualquier conjunto de pruebas que sea conforme con los interfaces descritos pueda ser utilizado junto con la plataforma. Estos interfaces de programación deberán proporcionar toda la funcionalidad necesaria para que todas las pruebas puedan adquirir la información que necesitan, al tiempo que llevan a cabo los pasos necesarios para desarrollar sus tareas.

- **Fraccionamiento de las Pruebas**: Como se puede ver en los capítulos 5 y 6, al aplicar la metodología propuesta en esta tesis se genera un elevado número de pruebas y ejecuciones de cada una de las pruebas, lo que hace que el análisis de una implementación se extienda en el tiempo de forma importante. Además, hemos visto también cómo varias de las pruebas a realizar necesitan de modificaciones constantes en la configuración de la implementación de IPsec, con el fin de evaluar el impacto de algún factor determinado en el rendimiento o la conformidad con el estándar.

  Dado que esta situación no es la más recomendable de cara a la usabilidad de la plataforma, se debe proporcionar a los usuarios la posibilidad de que el conjunto de pruebas que se llevarán a cabo se fraccionen en múltiples sesiones. El conjunto de estas sesiones abarcará todas las pruebas necesarias y finalmente proporcionará los resultados de la validación y evaluación de la implementación tal y como si se hubiese llevado a cabo todo el análisis en una única sesión.

  Adicionalmente, sería deseable que el usuario pudiese agrupar todas aquellas pruebas que requieren de configuraciones similares en la implementación de IPsec estudiada, lo que permitiría llevar a cabo todos esos análisis sin tener que alterar la configuración del dispositivo. De esta forma el usuario podría contar con "paquetes" de pruebas que pueden ejecutarse como un proceso por lotes, sin necesidad de estar alerta ante la necesidad de alterar la configuración de un dispositivo o equipo cada poco tiempo.

  La solución preliminar que se propone para integrar esta característica en la plataforma incluye el uso de tecnologías de almacenamiento (como por ejemplo, una Base de Datos) para almacenar los resultados de cada una de las sesiones parciales, y análisis detallados de los requisitos de cada prueba, con el objetivo de poder agrupar aquellas pruebas cuyos requisitos son compatibles, maximizando el número de

pruebas en cada grupo y minimizando el número de grupos.

- **Interpretación de Resultados**: Un último requisito de la plataforma de validación y evaluación remota es la inclusión de algún tipo de interpretación de los resultados obtenidos que facilite la comprensión por parte del usuario del significado práctico de los valores obtenidos tras los procesos de validación y evaluación.

  En este aspecto estamos trabajando en dos líneas diferentes: Por un lado se ofrecerán guías de configuración en la que se detallen las configuraciones óptimas desde dos puntos de vista diferentes: primando la interoperatividad y primando el rendimiento. De esta forma se proporcionará a los usuarios información acerca de cómo mejorar las prestaciones de su implementación de IPsec en estos dos aspectos.

  El otro aspecto en el que se trabaja es la comparación de los resultados de la implementación estudiada con respecto a otras implementaciones que hayan sido anteriormente analizadas en la plataforma, con el fin de que el usuario disponga de una visión de la posición relativa de su implementación en el conjunto de implementaciones IPsec del mercado. Para contar con esta información se utilizaría el soporte de almacenamiento de datos del que hablábamos al analizar el fraccionamiento de las pruebas.

  Sin embargo, esta información no podrá ser generada hasta que no se disponga de un número de implementaciones analizadas mínimo, que permita establecer comparaciones realistas y significativas.

Como podemos ver, la implementación de la arquitectura introduce nuevos retos que resolver, al tiempo que proporciona mejoras importantes con respecto a las pruebas atómicas en múltiples aspectos. Por estos motivos es muy importante dedicar el tiempo necesario a realizar un análisis y diseño riguroso, de forma que evitemos encontrarnos posteriormente con problemas que exijan deshacer parte del trabajo hecho. Como punto de partida se utilizará el diseño mostrado en la Figura 7.4.

En este esquema se muestra un interfaz Web que evita que los usuarios deban interactuar directamente con las pruebas en sí mismas. Los módulos de pruebas aparecen representados en la parte superior, existiendo un módulo de pruebas de validación de la conformidad, otro módulo de pruebas de evaluación del rendimiento, etc.... Un módulo de control se encarga de comunicar el interfaz web con el sistema de almacenamiento de resultados y con cada uno de los módulos de pruebas. Asimismo, el módulo de control también se comunica y configura el dispositivo encargado de generar artificialmente las condiciones de red definidas en los perfiles de tráfico[3], y con

[3]En esta implementación de la plataforma se ha optado por utilizar NIST Net ([124]) sobre un equipo con Linux que hace las funciones de puente entre las implementaciones.

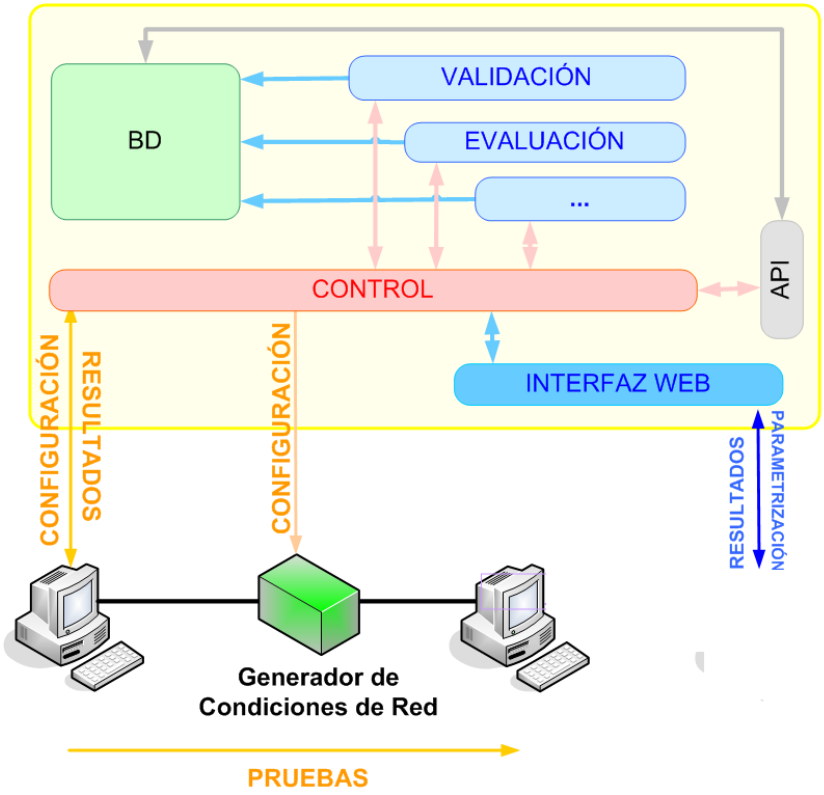

Figura 7.4: Esquema de la plataforma que implementará el conjunto de pruebas propuesto.

los sistemas que cuentan con la implementación de referencia, desde los que se lanzarán realmente las pruebas hacia la implementación que es objeto de la evaluación. Este módulo de control también es el responsable de gestionar el flujo de las pruebas que se están llevando a cabo, y comunicarse con el interfaz web para proporcionar información al usuario acerca del estado de las pruebas.

Todos los resultados y eventos de interés se almacenan en una Base de Datos, a la que es necesario acceder a través del módulo de control, o de cada una de los módulos de pruebas en sí mismas. Por último, aparece también un interfaz de programación (API) que permitiría a terceras aplicaciones acceder a los datos almacenados en la Base de Datos y al módulo de control, de forma que otros desarrollos puedan acceder tanto a las librarías de análisis (a través del módulo de control) como la información de la Base de Datos sin necesidad de utilizar el interfaz web.

Una de las principales ventajas de este diseño es su modularidad, ya que al independizar los módulos responsables de la comunicación con el usuario (interfaz web), configuración y control de la evaluación (control y cada módulo de pruebas adicional) y los sistemas que realmente llevan a cabo la batería de pruebas contra el sistema evaluado, es posible distribuir estas funcionalidades entre uno o varios sistemas diferentes, de acuerdo con los requisitos y características de cada usuario. Por ejemplo, en la Figura 7.5 podemos ver un ejemplo de despliegue para un usuario particular, en el que la implementación a evaluar se encuentra en el mismo equipo desde el que el usuario se conecta a la plataforma. Sin embargo, en la Figura 7.6 se propone una instalación en la que las funciones se encuentran repartidas entre múltiples sistemas, con lo que la capacidad de generación de tráfico y de uso en paralelo de la plataforma es mayor.

Otro efecto de la modularidad de este diseño es que al independizar las funciones de interfaz y control de los sistemas que realmente llevan a cabo las pruebas, se da pie a que la utilización de la plataforma pueda llevarse a cabo simultáneamente por múltiples usuarios. Para esto es necesario disponer de una infraestructura de red capaz de soportar todo el tráfico de red que se genere y de suficientes implementaciones de IPsec de referencia como para poder establecer los suficientes canales seguros como sea necesario.

Por último, al separar los sistemas que llevan a cabo las pruebas de los módulos que interactúan con el usuario, se ha eliminado uno de los potenciales problemas a los que nos enfrentábamos: la utilización de la red por parte del usuario para estudiar el rendimiento de la propia red. Con este diseño podemos observar que existen tres tipos de tráfico diferente:

1. **Tráfico del usuario al interfaz**: este tráfico se corresponde con el existente entre el interfaz web de la plataforma y el equipo desde el que el usuario hace uso de la mima. Es el que se genera al configurar

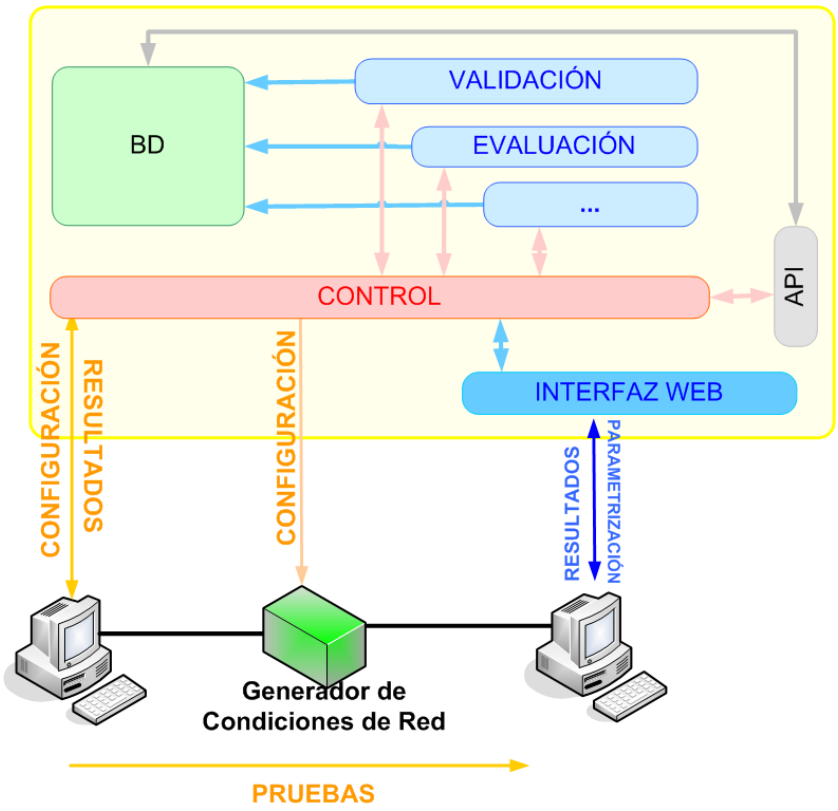

Figura 7.5: Esquema de utilización de la plataforma en el que un usuario selecciona las pruebas a llevar a cabo en la implementación de IPsec existente en el mismo equipo que utiliza para conectarse al interfaz web de la plataforma

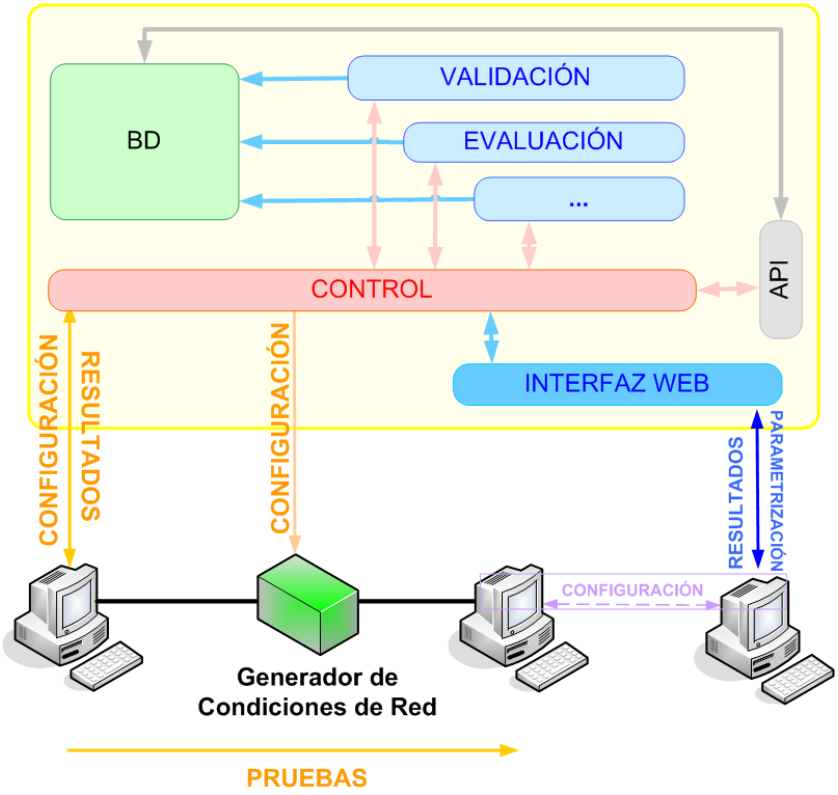

Figura 7.6: Esquema de utilización de la plataforma en el que un usuario selecciona las pruebas a llevar a cabo en la implementación de IPsec existente en otro dispositivo

las pruebas que se llevarán a cabo y al recibir los resultados de dichas pruebas.

2. **Tráfico del módulo de control a los sistemas de pruebas**: este tráfico es el que permite a la plataforma configurar los diferentes sistemas que se utilizarán para llevar a cabo las pruebas, así como iniciar y finalizar cada una de las pruebas individuales.

3. **Tráfico entre los sistemas de pruebas y la implementación evaluada**: este tráfico es el que se genera para la realización de cada una de las pruebas que se llevan a cabo.

Al independizar los módulos que generan estos tres tipos de tráfico es posible "aislar" estos canales de comunicaciones, evitando que, por ejemplo, el tráfico generado por el usuario con el interfaz de la plataforma interfiera con el desarrollo de las pruebas (como puede verse en la Figura 7.6. Adicionalmente, este aislamiento actúa como un mecanismo de seguridad que permite que en caso de que la comunicación entre el usuario y la plataforma se interrumpa (debido, por ejemplo, a la realización de pruebas con características de IPsec que no están soportadas en la implementación del usuario), esto no origine pérdida de datos alguna, ya que la comunicación entre los sistemas de pruebas y el módulo de control es independiente del sistema del usuario. De esta forma el usuario podrá acceder posteriormente al interfaz de la plataforma para consultar los resultados almacenados a la finalización de las pruebas, independientemente de si la comunicación entre su equipo y la plataforma se ha interrumpido o no.

7.4. Conclusiones

Como se ha podido ver en este apartado, las implementaciones desarrolladas en esta tesis han seguido dos caminos diferenciados: por un lado se han desarrollado pruebas atómicas que sirviesen de evaluación al conjunto de pruebas generado a partir de la aplicación de la metodología a la arquitectura IPsec; por otro lado se ha iniciado el desarrollo de una plataforma de validación y evaluación remota de implementaciones IPsec, en la que todas las aportaciones de esta tesis interoperen y ofrezcan resultados conjuntos.

Las pruebas atómicas se han desarrollado teniendo en mente que debían ser una evaluación para los métodos de medición y evaluación propuestos en la metodología, y que debían ser completamente funcionales. El hecho de representar una evaluación para métodos propuestos desde un plano más cercano al análisis teórico ha significado que al implementar cada una de las pruebas se han encontrado problemas y dificultades que se han solucionado bien replanteando la estrategia de la prueba en cuestión, bien ofreciendo soluciones novedosas a los problemas que se planteaban. Por otro lado, al

ser estas pruebas completamente funcionales es posible obtener información fiable acerca de una implementación de IPsec utilizando dichas pruebas.

El conjunto de las pruebas atómicas abarca completamente el conjunto definido en el capítulo 6, por lo que dicho conjunto de pruebas representa la primera implementación funcional de la metodología propuesta. Sin embargo, la implementación llevada a cabo en las pruebas atómicas carece de facilidad de uso para los usuarios, al tiempo que no presentan optimización y gestión adecuada de los recursos y librerías que utilizan.

Para paliar estos problemas y dar respuesta a otros requisitos, la siguiente implementación del conjunto de pruebas se ha llevado a cabo mediante una plataforma de validación y evaluación remota de implementaciones de IPsec. Desde esta plataforma es posible realizar análisis completos de una implementación de IPsec, de forma remota incluso, obteniendo la misma información que al utilizar las pruebas atómicas, aunque con mayor facilidad de uso.

El desarrollo de esta plataforma ha tenido que hacer frente a nuevos retos y dificultades, algunos de los cuales han sido ya abordados durante el diseño de la plataforma mientras que otros se han manifestado a la hora de trasladar las pruebas atómicas a una plataforma que integrase todas ellas. Utilizando un diseño que independiza las labores de interfaz, motor de pruebas y control ha sido posible proporcionar una solución elegante, escalable y con mecanismos para preservar la información en determinados casos de error.

Capítulo 8

Evaluación de la Tesis

En este capítulo se procederá a evaluar la presente tesis, analizando el grado de consecución de los objetivos planteados en el capítulo 1. Este estudio se llevará a cabo en dos fases: en la primera de ellas, desarrollada en la sección 8.1, se analiza cada uno de los objetivos propuestos para esta tesis, evaluando su grado de cumplimiento. La segunda fase, que se llevará a cabo en la sección 8.2, se centra en comprobar los resultados de someter a una implementación de IPsec a un conjunto de pruebas generado a partir de la metodología propuesta, estudiando los resultados obtenidos.

8.1. Evaluación de la tesis

Con el fin de evaluar el nivel de satisfacción de los objetivos propuestos en la presente tesis es necesario analizar, para cada uno de los que se propusieron, el grado de cumplimiento. En esta sección se llevará a cabo dicho análisis, en el que uno a uno se estudiará en qué medida la tesis ha cumplido con cada objetivo, analizando si la solución encontrada es susceptible de mejora.

8.1.1. Identificación de los parámetros de conformidad con el estándar

El primero de los objetivos propuestos en esta tesis es la identificación, a partir del estudio de especificaciones de protocolos y arquitecturas de seguridad estandarizados, de aquellos parámetros que condicionan el que una implementación sea conforme al estándar o no. La complejidad de los protocolos y arquitecturas de seguridad (tanto por la cantidad de protocolos y herramientas que la componen, como por su estructura modular en la que gran cantidad de esos componentes pueden ser reemplazados por otros), hace que sea muy complicado definir unos parámetros mínimos que todas

las implementaciones debieran cumplir para poder interoperar.

El análisis de los parámetros de conformidad con el estándar que es necesario evaluar parte de los propios documentos que definen las especificaciones de cada protocolo y componente de estos protocolos, en los que se encuentran recogidos los requisitos definidos por el organismo que estandarice el protocolo en cuestión, relativos a la conformidad con el estándar de cada componente. Este análisis, aunque válido, tiene el problema de que únicamente comprobaría un conjunto mínimo de requisitos para llevar a cabo el desarrollo de los protocolos. Sin embargo, los protocolos de seguridad dependen también en amplio grado de la utilización de las herramientas criptográficas (definidas también en otros documentos que forman parte del estándar, como por ejemplo [54], donde se definen y amplían las suites criptográficas que pueden y deben utilizarse con el protocolo ESP), al igual que lo hace de los mecanismos de autenticación y de gestión de claves que se integran en dichos protocolos. El motivo por el que en los documentos del estándar únicamente se hace referencia a la ejecución de los protocolos en los apartados de conformidad se debe a que dichos requisitos son los mínimos para que una implementación pueda establecer canales seguros con otro dispositivo semejante. Sin embargo, la interoperatividad con otras implementaciones de otros fabricantes no se pone a prueba en ningún momento.

Adicionalmente, los requisitos que se establecen para la conformidad resultan ser en muchos casos unos mínimos que, si bien aseguran que una implementación que cumpla con ellos no tendrá un comportamiento errático en cuanto al envío y recepción de mensajes pertenecientes a cada uno de los protocolos, no garantizan en ningún caso que dicha implementación sea capaz de recuperarse de la recepción de mensajes mal formados o incorrectos, ya que el estándar únicamente define que la implementación aceptará o no aceptará dicho mensaje. Esto ha llevado a que algunas implementaciones sean conformes con lo descrito en el estándar ya que, al recibir un mensaje mal formado la implementación interrumpe y destruye los canales seguros establecidos, como ocurre con la implementación de IPsec FreeS/WAN ([21]).

Dado que este comportamiento no es el aconsejable, en esta tesis se han diseñado las pruebas de tal forma que se identifica no sólo si una implementación es capaz de identificar un mensaje erróneo o mal formado, sino también si posteriormente la implementación es capaz de recuperarse del error y continuar operando sin que sea necesario interrumpir las comunicaciones que dicha implementación protege.

Por todos estos motivos, el análisis de los requisitos de conformidad necesarios para los protocolos y arquitecturas de seguridad llevado a cabo en esta tesis no sólo recoge la correcta ejecución del protocolo, sino que incluye el uso correcto de las herramientas criptográficas, mecanismos de autenticación y de gestión de claves, así como otras características que, aunque

inicialmente opcionales, deben ser evaluadas en el caso de utilizar arquitecturas de red concretas (como en el caso de NAT traversal en IPsec).

Adicionalmente el análisis llevado a cabo para identificar estos parámetros que es necesario evaluar, proporciona una lista de aspectos que deben ser evaluados, junto con un análisis acerca de los problemas que pueden surgir al validar esos parámetros y proponiendo posibles soluciones para superar dichos problemas.

Gracias a este análisis hemos podido constatar cómo se ha proporcionado una aproximación teórica a problemas que pueden surgir al llevar a cabo la validación de la implementación, tanto de un punto de vista teórico (como por ejemplo, la necesidad de disponer de una implementación de referencia en la que basar nuestro análisis) como práctico (la necesidad de enviar mensajes mal formados y llevar a cabo operaciones no permitidas según el estándar). Como consecuencia, en esta tesis se han propuesto soluciones novedosas para la validación de la implementación de herramientas manteniéndose en los límites marcados para la metodología. Por ejemplo, cabe citar el uso de la transitividad para evaluar la implementación de herramientas criptográficas en implementaciones que son tratadas como cajas negras, y a las que no podemos aplicar los vectores de pruebas propuestos en las especificaciones de dichas herramientas.

Por otro lado, el hecho de que las medidas propuesta hayan podido ser llevadas a la práctica en forma de pruebas atómicas que desarrollan análisis concretos, nos indica que el estudio llevado a cabo acerca del modo en el que implementar dichos análisis ha dado sus frutos, aplicando la metodología en un conjunto de pruebas cuya implementación es viable y sus resultados ofrecen información concreta acerca de aspectos determinados, aspectos que son los que determinan el grado de conformidad con el estándar de la implementación de IPsec y sus posibilidades de interoperatividad.

8.1.2. Identificación de los parámetros de rendimiento

Otro de los objetivos de esta tesis es la identificación de parámetros que afecten al rendimiento de los protocolos y arquitecturas de seguridad. En este caso el problema era diferente, ya que, aunque existen múltiples propuestas y herramientas para obtener información acerca del rendimiento de los sistemas de comunicaciones, dichos sistemas no aportan información útil al introducir mecanismos de seguridad en dichos sistemas, como se vio en el capítulo 2.

Aunque en teoría hubiese sido posible llevar a cabo un estudio del rendimiento de las implementaciones de protocolos de seguridad utilizando metodologías de análisis de rendimiento tradicionales, hemos visto cómo los análisis realizados indican que las diferencias entre las comunicaciones cuando en el proceso de transmisión se integran mecanismos de seguridad no

afectan únicamente a las medidas de rendimiento, sino que también afectan al conjunto de aspectos que es necesario evaluar.

En esta tesis se ha llevado a cabo un análisis exhaustivo de las diferencias entre el rendimiento de los sistemas de comunicaciones sin ningún tipo de mecanismo de seguridad integrado, y el rendimiento de los mismos sistemas al utilizar un protocolo de seguridad como SSL/TLS o una arquitectura de seguridad como IPsec. A partir de las diferencias obtenidas ha sido posible analizar cuáles son los recursos que se ven más afectados al utilizar este tipo de soluciones de seguridad, y en qué afectan estos recursos al rendimiento del sistema.

Una aportación importante de esta tesis es la identificación del establecimiento de conexión como uno de los componentes que más recursos requieren, especialmente ciclos de proceso en la CPU y espacio de memoria, al contrario que en una comunicación sin mecanismos de seguridad, en la que el establecimiento de conexión tiene unos requisitos mucho menores. Este tipo de diferencias sutiles han sido detectadas durante la fase de análisis del rendimiento de los protocolos de seguridad, lo que nos ha llevado a evitar replicar una metodología tradicional de análisis del rendimiento de redes de comunicaciones, haciendo hincapié en las características propias de los protocolos y arquitecturas de seguridad.

Por otro lado, y fruto de este mismo enfoque, ha surgido la necesidad de utilizar perfiles de tráfico para evaluar correctamente el rendimiento de las implementaciones de los protocolos de seguridad, ya que una de las principales características identificadas es la diferencia entre los requisitos para cifrar y descifrar, por lo que, desde el punto de vista del rendimiento, no es lo mismo recibir tráfico a una velocidad determinada que generar tráfico a esa misma velocidad. Este aspecto, que en las comunicaciones sin mecanismos de seguridad no afecta a los resultados de los análisis de rendimiento, es crucial para las implementaciones de los protocolos de seguridad.

Así pues, la conjunción del estudio llevado a cabo para la identificación de los parámetros de rendimiento que es necesario evaluar, y el análisis posterior de los métodos que pueden emplearse para llevar a cabo dicha evaluación, conforman un estudio en profundidad acerca de la evaluación del rendimiento en implementaciones de protocolos y arquitecturas de seguridad.

Al igual que ocurría anteriormente, el haber podido trasladar este análisis a implementaciones prácticas con las que se llevan a cabo pruebas necesarias indican que la evaluación del rendimiento según el análisis propuesto es factible, obteniéndose resultados que permiten conocer en detalles cuál será el comportamiento de la implementación evaluada al llevar a cabo su función, y también cómo responderá a denegaciones de servicio (accidentales o provocadas).

8.1.3. Establecimiento de un marco de análisis y desarrollo para protocolos y arquitecturas de seguridad

El análisis llevado a cabo ha permitido estudiar las características y particularidades de los protocolos y arquitecturas de seguridad, tanto en lo referente a conformidad e interoperabilidad como en lo tocante al análisis del rendimiento. Muchos de los conocimientos adquiridos y estudios realizados son una valiosa base para el desarrollo de posteriores aplicaciones similares a otros protocolos y arquitecturas de seguridad, generándose así un marco de trabajo que englobe tanto las labores de análisis que se han llevado a cabo como los desarrollos que han surgido de esta tesis.

El análisis de los aspectos que es necesario evaluar en lo tocante a conformidad y rendimiento que se ha llevado a cabo en la presente tesis proporciona unos fundamentos sobre los que construir y desarrollar en profundidad un estudio más detallado y amplio, en el que se abarquen otras arquitecturas y protocolos de seguridad, así como los problemas de interoperatividad y rendimiento que se presentan y los orígenes de dichos problemas. De esta forma se podrá construir una base común en la que aparezcan reflejadas todas las características sobre las que llevar a cabo los análisis, de forma que la mejora de esta base común permita hacer avanzar los conocimientos y desarrollos sobre la interoperatividad y rendimiento de implementaciones de protocolos y arquitecturas de red.

En este aspecto la presente tesis ha contribuido a este objetivo trabajando en dos frentes diferentes pero complementarios:

- Por un lado, los estudios de rendimiento e interoperatividad se basan en características de los protocolos y arquitecturas de seguridad que los distinguen de los protocolos de red sin seguridad incluida. Posteriormente, esas características se reflejan en la metodología que se ha propuesto en el capítulo 5 y se ha desarrollado en el capítulo 6 para la arquitectura IPsec, pero el núcleo abstracto del estudio puede ser aplicado a la gran mayoría de los protocolos y arquitecturas de seguridad.

  Adicionalmente, los análisis acerca de métodos y mecanismos que permitan salvar las dificultades que puedan surgir al implementar los métodos de validación y evaluación propuestos, se han llevado a cabo evitando proponer soluciones particulares para una arquitectura o protocolo concreto, resultando por lo tanto que los avances logrados mediante dichos análisis pueden ser aplicables al resto de protocolos y arquitecturas de seguridad.

- Por otro lado, al implementar las pruebas surgidas de la aplicación de la metodología a IPsec se ha independizado el motor de prueba de las librerías necesarias para trabajar con IPsec, resultando de este método

de desarrollo unas librerías de validación y evaluación de protocolos de seguridad que, mediante una adecuada integración y desarrollo de puntos de entrada, permitiría su utilización con otros protocolos de seguridad.

Este aspecto seguirá mejorándose junto con el desarrollo de la plataforma de validación y evaluación remota, para permitir que el trabajo desarrollado pueda ser utilizado para obtener información de otros protocolos y arquitecturas.

Por todos estos motivos podemos decir que la presente tesis ha cumplido con el objetivo que se marcó en sus inicios, al haber proporcionado los medios necesarios para que los conocimientos y desarrollos fruto de la labor realizada puedan ser reutilizados y mejorados en el futuro.

8.1.4. Metodología de validación y evaluación de implementaciones de protocolos de seguridad

La metodología de validación y evaluación de implementaciones de protocolos de seguridad parte de los análisis anteriores para, utilizando los métodos de evaluación propuestos en cada uno de los análisis, construir una metodología que permita evaluar la conformidad con el estándar y evaluar el rendimiento de implementaciones de protocolos de seguridad.

La metodología propuesta ofrece un conjunto estructurado de guías y orientaciones enfocadas a validar aspectos concretos de la conformidad y evaluar el rendimiento de la implementación del protocolo de seguridad que se desea analizar. Para cada aspecto que deba ser evaluado se analizan qué aspectos deben tenerse en cuenta a la hora de llevar a cabo esa evaluación, los mecanismos y técnicas que pueden utilizarse para las tareas de captura de datos, los pasos a llevar a cabo para la obtención y procesado de los resultados que se obtengan. Este análisis propuesto en la metodología permite recabar información acerca de aquellos aspectos que se han identificado como cruciales para asegurar la interoperabilidad entre diferentes implementaciones de los protocolos de seguridad. Por lo tanto, la metodología también permite obtener información que puede utilizarse para evaluar el nivel de interoperabilidad de dos implementaciones independientes.

Adicionalmente, la estructura modular (en fases) de la metodología permite su constante revisión y actualización para mantenerla al día e incorporar modificaciones y actualizaciones a la misma. Del mismo modo, en algunos aspectos (como por ejemplo, la conformidad de la implementación de herramientas criptográficas) la metodología es flexible, de tal forma que permite abarcar los diferentes protocolos y arquitecturas de seguridad existentes, sin cerrarse a ninguno ni entrar en detalles específicos de alguno de ellos.

En cuanto a la evaluación del rendimiento de una implementación IPsec, la metodología propone baterías de pruebas que permiten identificar cuál es el coste de utilizar IPsec para proteger el tráfico de nuestra red (en latencia de la red y ancho de banda efectivo del que se puede disponer), cómo afecta dicha penalización según la configuración de IPsec que se utilice, cómo afecta el tráfico de red a la capacidad de establecer nuevos túneles criptográficos (y por lo tanto, a la capacidad de crecimiento de la red utilizando la misma implementación de IPsec) y qué fases de IPsec representan o pueden representar un cuello de botella en el rendimiento.

Los análisis que se proponen son exhaustivos, en tanto que todos los aspectos que pueden afectar al rendimiento son tenidos en cuenta para obtener la información de rendimiento. Aunque este enfoque eleva la cantidad total de pruebas que se generan al aplicar esta metodología a un protocolo o arquitectura de seguridad concreto, el resultado final es un análisis pormenorizado del que se puede extraer cuál la influencia exacta de cada característica en el rendimiento que la implementación está ofreciendo.

Al igual que hemos comentado para la validación de la implementación, la modularidad de los conjuntos de pruebas descritos para la evaluación del rendimiento hace que sea posible mantener la metodología actualizada o incluso adaptada a necesidades particulares.

Por lo tanto, la metodología propuesta es capaz de presentar métodos estructurados para evaluar los diferentes aspectos de conformidad y rendimiento que deben ser analizados en una implementación de un protocolo o arquitectura de seguridad.

8.1.5. Aplicación de la metodología a la arquitectura de seguridad IPsec

Una vez definida la metodología en el capítulo 5, se ha procedido a verificar la aplicabilidad de la metodología a una arquitectura de seguridad concreta; en este caso, IPsec. En el capítulo 6 se ha llevado a cabo una aplicación de las directrices formuladas en la metodología a la arquitectura de seguridad, lo que ha generado un conjunto de pruebas destinados a llevar a cabo los procesos de validación de la conformidad y evaluación del rendimiento.

A la hora de aplicar la metodología se ha podido comprobar cómo ha sido posible desarrollar conjuntos de pruebas que forman un conjunto completo respecto de las características que toda implementación de IPsec debe desarrollar para ser conforme con el estándar. Para ello se han analizado tal y como especifica la metodología todos y cada uno de los componentes de la arquitectura: protocolos, algoritmos criptográficos, etc. . . . y se han diseñado pruebas que permitiesen obtener información acerca de cómo se encuentran implantados cada uno de esos componentes en la implementación que se

está analizando.

En cuanto a la evaluación del rendimiento, ha sido posible definir conjuntos de pruebas que analizan los diferentes aspectos del rendimiento de protocolos y arquitecturas de seguridad que es interesante conocer (según el análisis llevado a cabo en el capítulo 4) aplicados a la idiosincrasia de la arquitectura de seguridad IPsec. Estos conjuntos de pruebas permiten evaluar el rendimiento de la implementación cuando se utilizan determinadas combinaciones de protocolos, algoritmos y mecanismos de seguridad, por lo que permiten obtener una visión general del rendimiento global, así como detalles específicos acerca del rendimiento en condiciones concretas.

Por último, cabe destacar que en la aplicación de la metodología a IPsec se ha llevado a cabo también un proceso de definición de perfiles de utilización de la implementación que es interesante utilizar a la hora de obtener información realista acerca del rendimiento de la implementación. Algunos de estos perfiles tienen por objetivo la identificación de valores "representativos" del rendimiento de la implementación (por ejemplo, el máximo ancho de banda que se puede proteger), mientras que otros están orientados a la recogida de información acerca del rendimiento que la implementación puede ofrecer en condiciones realistas (por ejemplo, utilizando comunicaciones con TCP, UDP e ICMP, y existiendo un 1 % de paquetes descartados en la red).

8.1.6. Desarrollo de una plataforma de validación y evaluación remota de implementaciones de IPsec

Los desarrollos llevados a cabo durante el desarrollo de la presente tesis han seguido dos enfoques diferentes, aunque complementarios. Por un lado se han desarrollado las pruebas atómicas que implementan cada una de las pruebas y análisis individuales descritos en la aplicación de la metodología a la arquitectura IPsec; por otro lado, el diseño y desarrollo de una plataforma de validación y evaluación remota pretende ser el hito final en el que interoperan todas las aportaciones de la presente tesis.

Las pruebas atómicas desarrolladas como paso inicial en la implementación del conjunto de pruebas propuesto han resultado ser un método de evaluación del análisis llevado a cabo para los factores de conformidad y de rendimiento, así como para los métodos de medición y evaluación que se habían propuesto en dichos análisis. En esta faceta de sistema de autocontrol para las propuestas que se han realizado en esta tesis, las pruebas atómicas han resultado ser un método eficaz a la par que sencillo de llevar a cabo.

Por otro lado, las pruebas atómicas también han servido para establecer un punto de partida para la implementación de la plataforma de validación y evaluación remota de implementaciones de IPsec, desde el que llevar a

cabo el diseño y establecer diferencias entre ambas implementaciones. Este estudio de las diferencias entre las pruebas atómicas y la plataforma de validación e implementación remota es una ventaja fundamental de cara al análisis de requisitos e identificación de las dificultades con las que podremos encontrarnos.

Sin embargo, no debemos olvidar que las pruebas atómicas también son un desarrollo de las pruebas generadas al aplicar la metodología a implementaciones IPsec, y como tal, son capaces de analizar una implementación de IPsec y ofrecernos información acerca de su conformidad con el estándar y del rendimiento que dicha implementación puede ofrecer, como se verá en la sección 8.2.

En cuanto a la plataforma de validación y evaluación remota de implementaciones IPsec, su desarrollo ha permitido contar con un entorno amigable que sirva para llevar a cabo la validación de la conformidad de una implementación de IPsec, al tiempo que evaluamos el rendimiento que dicha implementación es capaz de ofrecer. Durante el proceso de desarrollo de la plataforma se ha llevado a cabo una labor de análisis que ha permitido, por un lado, independizar el motor de pruebas de la implementación del protocolo de seguridad,y por otro, definir mecanismos para que la plataforma pueda integrarse con otras herramientas que maximicen el valor de la información recabada en la plataforma.

Las implementaciones de los conjuntos de pruebas que se han desarrollado, tanto en forma de pruebas atómicas como siendo parte de la plataforma, han sido utilizadas en varios proyectos de investigación, permitiendo obtener información acerca de la conformidad y el rendimiento de varias implementaciones de IPsec de diversos fabricantes. Asimismo, diversos grupos de investigación de centros internacionales han mostrado su interés por la plataforma de validación y evaluación remota.

8.2. Evaluación de Implementaciones de IPsec

Con el fin de conocer cuáles son los resultados que ofrece la metodología al analizar una implementación de IPsec, en esta sección presentamos los resultados obtenidos al aplicar la metodología, haciendo uso de las pruebas atómicas y de la plataforma de evaluación descritas en la sección 7.2. La implementación analizada ha sido OpenS/WAN v2.4.4 , ejecutándose sobre un sistema operativo Linux. Las características completas del sistema evaluado pueden verse en la Tabla 8.1.

Los resultados obtenidos al aplicar las pruebas que desarrollan la metodología para la arquitectura IPsec se desarrollarán en los siguientes apartados. En ellos se hará referencia a los nombres de las pruebas atómicas tal y como se declararon en el capítulo 7.

Tabla 8.1: Especificaciones de la implementación IPsec evaluada

Sistema Operativo	Linux (kernel 2.6.15)
Procesador	AMD Athlon 550 MHz
Memoria	384 MB
Implementación de IPsec	OpenS/WAN 2.4.4
Herramientas Criptográficas	Compiladas en el núcleo del sistema operativo: HMAC-MD5, HAC-SHA1, MD4, MD5, SHA256, SHA384, SHA512, DES, 3DES, Blowfish, Twofish, Serpent, AES, CAST5, CAST6, TEA, ARC4
Versión de IPsec	IPsec versión 1
Tecnología de Red	Fast Ethernet dedicada, con enlace directo entre las implementaciones

Tabla 8.2: Resultados de validación de las herramientas criptográficas utilizadas en ESP

Prueba	**Resultado**
`EspNull`	OK
`Esp3Des`	OK
`EspAES`	OK

8.2.1. Validación de la conformidad

Como podemos ver, la implementación evaluada únicamente presenta problemas de conformidad al utilizar AES-XCBC-MAC-96, ya que dicho algoritmo no se encuentra disponible en dicha implementación. Todo el resto de pruebas realizadas ofrecen resultados positivos, tanto para la Fase 1 como la Fase 2 y en ESP. Dado que el actualmente AES-XCBC-MAC-96 es opcional para IPsec v1, **las herramientas criptográficas en OpenS/WAN son conformes a lo especificado en el estándar**.

También podemos observar cómo, en lo tocante al desarrollo de los protocolos, la implementación evaluada no presenta ningún problema en ninguno de los protocolos, modos de funcionamiento ni opciones disponibles. Por lo tanto, **la implementación de los protocolos también es conforme a los estándares**.

Tabla 8.3: Resultados de validación de las herramientas criptográficas utilizadas en la Fase 1 de IKE

Prueba	Resultado
Fase13DesMD5	OK
Fase13DesSHA1	OK
Fase13DesAES	No Disponible
Fase1AesSHA1	OK
Fase1AesAES	No Disponible

Tabla 8.4: Resultados de validación de las herramientas criptográficas utilizadas en la Fase 2 de IKE

Prueba	Resultados
Fase23DesMD5	OK
Fase23DesSHA1	OK
Fase23DesAES	No Disponible
Fase2AesSHA1	OK
Fase2AesAES	No Disponible

Como podemos ver, OpenS/WAN no permite la utilización de certificados X.509 para llevar a cabo los procesos de autenticación. Dicha funcionalidad está soportada mediante parches que no fueron aplicados a la implementación evaluada, por lo que no es posible evaluar esta característica. Por lo tanto, y dado que la **autenticación** mediante certificados X.509 es un requisito para la arquitectura de seguridad IPsec, **OpenS/WAN no es conforme al estándar en este aspecto**.

En cuanto a la gestión de claves, OpenS/WAN no presenta ningún problema al controlar las renovaciones de claves criptográficas, tanto en la Fase 1 como en la Fase 2.

Por último, las características adicionales que es posible que sean necesarias para utilizar IPsec en nuestra topología de red están irregularmente implementadas en OpenS/WAN. **Mientras que NAT traversal no presenta problema alguno, la compresión de datos en la capa IP presenta problemas** al interoperar con otras implementaciones. Por último, la notificación de la congestión de la red no se encuentra implementada en OpenS/WAN.

Tabla 8.5: Resultados de validación del desarrollo de los protocolos en modo túnel

Prueba	Resultado
TunelMMQMESP	OK
TunelMMQMAH	OK
TunelMMQMESPPFS	OK
TunelMMQMAHPFS	OK
TunelAMQMESP	OK
TunelAMQMAH	OK
TunelAMQMESPPFS	OK
TunelAMQMAHPFS	OK

Tabla 8.6: Resultados de validación del desarrollo de los protocolos en modo transporte

Prueba	Resultados
TranspMMQMESP	OK
TranspMMQMAH	OK
TranspMMQMESPPFS	OK
TranspMMQMAHPFS	OK
TranspAMQMESP	OK
TranspAMQMAH	OK
TranspAMQMESPPFS	OK
TranspAMQMAHPFS	OK

8.2.2. Evaluación del rendimiento

A la hora de evaluar el rendimiento, se han incorporado a los resultados los valores obtenidos al utilizar los perfiles en los que la red se encuentra saturada. Estos valores aparecen en las siguientes tablas con el sufijo `-sat` para su fácil identificación.

Tabla 8.7: Resultados de validación de los mecanismos de autenticación

Prueba	Resultado
PskMd5MM	OK
PskMd5AM	OK
PskSha1MM	OK
PskSha1AM	OK
RsaMM	OK
RsaAM	OK
X509MM	No Disponible
X509AM	No Disponible

Tabla 8.8: Resultados de validación de la gestión de claves

Prueba	Resultado
Fase1Tiempo	OK
Fase2Tiempo	OK
Fase1Trafico	OK
Fase2Trafico	OK

Tabla 8.9: Resultados de validación de otras características adicionales

Prueba	Resultado
NAT	OK
IPComp	ERROR
ECN	No Disponible

Tabla 8.10: Resultados de evaluación del ancho de banda.

Prueba	Configuración	Resultado
BWTMax	ESP NULL	95 Mbps
BWTMax	ESP 3DES - MD5	65 Mbps
BWTMax	ESP 3DES - SHA1	50 Mbps
BWTMax	AH MD5	80 Mbps
BWTMax	AH SHA1	70 Mbps
BWTMax-sat	ESP NULL	35 Mbps
BWTMax-sat	ESP 3DES - MD5	25 Mbps
BWTMax-sat	ESP 3DES - SHA1	22 Mbps
BWTMax-sat	AH MD5	30 Mbps
BWTMax-sat	AH SHA1	28 Mbps
Trad	ESP NULL	65 Mbps
Trad	ESP 3DES - MD5	50 Mbps
Trad	ESP 3DES - SHA1	42 Mbps
Trad	AH MD5	60 Mbps
Trad	AH SHA1	56 Mbps
TCPAsimIn	ESP NULL	52 Mbps
TCPAsimIn	ESP 3DES - MD5	44 Mbps
TCPAsimIn	ESP 3DES - SHA1	41 Mbps
TCPAsimIn	AH MD5	49 Mbps
TCPAsimIn	AH SHA1	47 Mbps
TCPAsimOut	ESP NULL	54 Mbps
TCPAsimOut	ESP 3DES - MD5	47 Mbps
TCPAsimOut	ESP 3DES - SHA1	44 Mbps
TCPAsimOut	AH MD5	52 Mbps
TCPAsimOut	AH SHA1	48 Mbps
TCPSim	ESP NULL	69 Mbps
TCPSim	ESP 3DES - MD5	55 Mbps
TCPSim	ESP 3DES - SHA1	48 Mbps
TCPSim	AH MD5	63 Mbps
TCPSim	AH SHA1	60 Mbps
TCPSim-sat	ESP NULL	28 Mbps
TCPSim-sat	ESP 3DES - MD5	22 Mbps
TCPSim-sat	ESP 3DES - SHA1	20 Mbps
TCPSim-sat	AH MD5	20 Mbps
TCPSim-sat	AH SHA1	19 Mbps

Tabla 8.11: Resultados de evaluación del número máximo de asociaciones de seguridad

Nombre	Resultado
`MaxSA`	163 Asociaciones de Seguridad

Tabla 8.12: Resultados de establecimiento de asociaciones de seguridad (Extracto)

Configuración	Mbps / SA	Conexiones / seg.
DH2, IKE: 3DES MD5, ESP: 3DES SHA1	0,5 Mbps	1
DH2, IKE: 3DES MD5, ESP: 3DES SHA1	1 Mbps	1
DH2, IKE: 3DES MD5, ESP: 3DES SHA1	2 Mbps	0,5
DH2, IKE: 3DES MD5, ESP: 3DES SHA1	3 Mbps	0,3
DH2, IKE: 3DES MD5, ESP: 3DES SHA1	4 Mbps	0,2
DH2, IKE: 3DES MD5, ESP: 3DES SHA1	5 Mbps	0,2
DH2, IKE: AES SHA1, ESP: AH SHA1	0,5 Mbps	1,4
DH2, IKE: AES SHA1, ESP: AH SHA1	1 Mbps	1
DH2, IKE: AES SHA1, ESP: AH SHA1	2 Mbps	0,8
DH2, IKE: AES SHA1, ESP: AH SHA1	3 Mbps	0,5
DH2, IKE: AES SHA1, ESP: AH SHA1	4 Mbps	0,5
DH2, IKE: AES SHA1, ESP: AH SHA1	5 Mbps	0,4

Al analizar el rendimiento de la implementación de IPsec, vemos que, sin aplicar ningún tipo de protección al tráfico, la implementación únicamente es capaz de transmitir 95 Mbps. Este bajo resultado hace que el resto de valores se vean afectados y se obtengan velocidades de transmisión en torno a los 50 Mbps, un 25 % del máximo teórico del canal. Asimismo, podemos ver cómo la saturación del canal de datos afecta al rendimiento que es capaz de ofrecer la implementación.

En cuanto a la capacidad de establecer asociaciones de seguridad, vemos que el número máximo de asociaciones que se pueden establecer es de 163. Sin embargo, si por los túneles criptográficos se transmite información, la velocidad a la que dichos túneles se pueden establecer es más lenta, llegándose a casos en los que hay que esperar hasta 5 segundos entre una conexión y otra.

Capítulo 9

Conclusiones

En este capítulo se procederán a presentar las conclusiones de la presente tesis. Estas conclusiones partirán de un análisis de las aportaciones novedosas de la tesis, para continuar con un estudio de las futuras líneas de investigación a las que el desarrollo de esta tesis nos puede conducir.

9.1. Aportaciones de la tesis

Las principales aportaciones de la presente tesis se corresponden con los objetivos que se definieron al inicio de la misma: identificación y análisis de los parámetros que son relevantes para evaluar la conformidad con el estándar, identificación y análisis de los parámetros que resultan relevantes para evaluar el rendimiento de la implementación, establecimiento de un marco de análisis y desarrollo para protocolos de seguridad, propuesta de una metodología de validación y evaluación remota de implementaciones de protocolos de seguridad, aplicación de la metodología a la arquitectura de seguridad IPsec y desarrollo de una plataforma de validación y evaluación remota de implementaciones de IPsec.

A continuación desarrollaremos las conclusiones que se pueden obtener de cada uno de estos aspectos de la tesis por separado.

9.1.1. Análisis de conformidad

El análisis de los aspectos de la conformidad con las especificaciones del estándar que deben ser incluidos en una metodología que se ha llevado a cabo en el marco de esta tesis, ha ofrecido aportaciones interesantes, no sólo en el ámbito de los aspectos de un protocolo o arquitectura de seguridad que es necesario analizar, sino también en el ámbito de los métodos que deben utilizarse para llevar a cabo dichos análisis.

Por lo tanto, los análisis llevados a cabo se han compuesto de dos partes diferenciadas, aunque relacionadas:

- En una primera fase, las diferentes características de los protocolos de seguridad que definen la conformidad de una implementación con respecto al estándar que define dicho protocolo o arquitectura, han sido identificadas y agrupadas en "familias" de aspectos que facilitasen su posterior utilización.

- A continuación, los diferentes métodos para evaluar dichas características han sido revisados y su idoneidad para ser incluidos en la metodología ha sido evaluada. Dado el especial enfoque de la metodología propuesta muchos de los métodos de evaluación existentes no eran adecuados para utilizarse en dicha metodología, por lo que se han propuesto nuevos métodos alternativos que obtengan la información necesaria ajustándose a los requisitos de la metodología.

Este análisis de las características de los protocolos y arquitecturas de seguridad que es necesario evaluar ha permitido definir, de forma estructurada y fácilmente manejable, aquellos aspectos que son origen de los problemas de interoperatividad entre implementaciones de diferentes fabricantes. Adicionalmente, el análisis ha permitido que sea posible definir pruebas comunes para aquellos aspectos que pertenecen a la misma "familia", simplificándose enormemente tanto la identificación como la evaluación de las características que es necesario evaluar.

Por otro lado, la necesidad de proponer métodos de evaluación que se ajustasen a los requisitos de la metodología (especialmente, al requisito de operar con la implementación únicamente como si de una caja negra se tratase) ha hecho que en la mayoría de los casos se hayan propuesto nuevas técnicas de evaluación de los aspectos en cuestión.

Adicionalmente, el análisis de los problemas existentes entre las diferentes implementaciones de IPsec ha conducido a la identificación de ambigüedades en los requisitos mínimos definidos en los estándares de IPsec, por lo que los métodos de evaluación propuestos a partir de este análisis se basan en requisitos más estrictos que los definidos en el estándar, pero que permiten conocer cuál será el comportamiento de la implementación estudiada al operar con otras implementaciones.

Por todos estos motivos podemos decir que el análisis de los aspectos de conformidad con la arquitectura de seguridad IPsec ha servido para realizar aportaciones interesantes a los métodos de evaluación de características de conformidad. Asimismo, se han utilizado agrupaciones de estas características para ofrecer un manejo de estos parámetros que facilitase la posterior incorporación en la metodología.

9.1.2. Análisis de rendimiento

En cuanto al análisis de rendimiento, el trabajo llevado a cabo en esta tesis ha desarrollado un análisis en el que se estudian los factores de rendimiento que mayor influencia tienen durante el desarrollo de un protocolo o arquitectura de seguridad. Este análisis ha partido de un estudio particular, estudiando las características del protocolo TLS, para después generalizar los resultados obtenidos a cualquier protocolo o arquitectura de seguridad.

Una de las principales diferencias que presenta el análisis llevado a cabo con respecto a las metodologías de evaluación del rendimiento en comunicaciones sin seguridad integrada, es que, mientras que las últimas pueden incluir entre sus operaciones la extrapolación de resultados a partir de mediciones en condiciones que no representan la situación que se desea evaluar, en un protocolo o arquitectura de seguridad esto no es así, ya que el análisis ha demostrado cómo no es posible predecir el comportamiento de las diferentes arquitecturas al llevar a cabo operaciones criptográficas.

Por lo tanto, el análisis de rendimiento ha definido el conjunto de parámetros cuya evaluación nos proporcionará información completa acerca del rendimiento que se puede esperar de una implementación de un protocolo de seguridad. Adicionalmente, de esta relación de parámetros a evaluar, junto con el estudio de los mecanismos que pueden aplicarse para obtener la información requerida, ha surgido un conjunto de pruebas que permiten obtener la información requerida.

Adicionalmente al desarrollo del conjunto de pruebas a llevar a cabo, la otra gran aportación de este análisis ha sido la constatación de que, al utilizar un protocolo de seguridad, el tipo de tráfico que se protege afecta al rendimiento que se obtendrá. Por lo tanto, el rendimiento obtenido no será el mismo si se protege tráfico TCP en una única dirección que si lo que se protege es tráfico UDP en ambas direcciones. Igualmente, las condiciones en las que se encuentra la red (saturación, pérdida de tramas, retransmisión de paquetes, etc....) afecta a los resultados de rendimiento que se obtienen, por lo que se ha incluido en la definición de los diferentes perfiles de tráfico el estado de la red.

Esta aportación se ve reflejada en la tesis en la definición de múltiples perfiles de tráfico que deben ser utilizados para obtener distinto tipo de información de rendimiento en cada una de las pruebas resultantes del análisis. La ejecución de la misma prueba con diferente perfil de tráfico generará unos resultados diferentes, con un significado diferente en función de las características de tráfico y de la red que defina el perfil.

En conclusión, el análisis de los factores de rendimiento de un protocolo o arquitectura de seguridad ha proporcionado información acerca de la influencia de los mecanismos de seguridad utilizados por los protocolos y arquitecturas de seguridad en el rendimiento, al tiempo que ha proporcio-

nado los medios para llevar a cabo mediciones exhaustivas en las que poder obtener toda la información de rendimiento que resulta interesante conocer.

9.1.3. Marco de análisis y desarrollo para protocolos y arquitecturas de seguridad

Otro de los objetivos definidos para esta tesis es el establecimiento de un marco de análisis y desarrollo para protocolos y arquitecturas de seguridad. Este objetivo se fundamenta en la necesidad de llevar a cabo estudios similares a los que se han llevado a cabo durante el desarrollo de esta tesis, aplicando la metodología a otros protocolos o arquitecturas de seguridad, y ampliándola en caso necesario. Dado que en la presente tesis se han llevado a cabo análisis y desarrollos que operan con los propios fundamentos de los aspectos que se desea evaluar, la reutilización de dicho trabajo en estudios futuros es una posibilidad que permitirá concentrar esfuerzos en las características distintivas.

En la presente tesis el marco de análisis y desarrollo se ha visto refrendado por dos tipos de aportaciones:

- Por un lado, los análisis de los factores de conformidad y rendimiento que se han llevado a cabo tienen su fundamento en el análisis de protocolos y soluciones concretas, análisis del que posteriormente se extrapolan los factores que son comunes a la mayoría de protocolos y arquitecturas de seguridad.

  Esta extrapolación permite que cualquier análisis similar que se quiera llevar a cabo de una solución de seguridad particular pueda partir de la generalización llevada a cabo, con el consiguiente ahorro de tiempo y esfuerzo.

- Por otro lado, las implementaciones desarrolladas en esta tesis han proporcionado un conjunto de librerías de desarrollo en las que se pueden encontrar los métodos de medición definidos en la metodología, que han sido independizados de los procesos de generación de mensajes y establecimiento de túneles propios de IPsec. Cualquier otra propuesta que se base en estos métodos de captura de la información requerida podrá utilizar dichas librerías de desarrollo para evitar la duplicación de esfuerzos. Asimismo, dichas librerías podrán ser actualizadas y mejoradas según sea necesario.

Como vemos, el marco de análisis y desarrollo para protocolos de seguridad ha sido establecido, y las primeras aportaciones a dicho marco se han producido, tanto en el ámbito del análisis de los protocolos y arquitecturas de seguridad como en el desarrollo de herramientas para llevar a cabo los análisis propuestos.

9.1.4. Metodología de validación y evaluación remota de implementaciones de protocolos de seguridad

El desarrollo de una metodología de validación y evaluación que, a partir de los resultados de los análisis de conformidad y rendimiento obtenidos anteriormente, permitiera llevar a cabo un estudio en profundidad acerca de las capacidades de interoperatividad y rendimiento de las implementaciones de protocolos de seguridad representa la principal aportación de esta tesis.

Como hemos comentado, la metodología se basa en los análisis de factores de conformidad y de rendimiento llevados a cabo anteriormente, y de hecho utiliza sus resultados como base sobre la que definir el conjunto de líneas de actuación que conforman de la metodología: A partir de los estudios acerca de los métodos idóneos para llevar a cabo mediciones sobre cada uno de los aspectos concretos que se desean evaluar, tanto en relación con el rendimiento como en relación con la conformidad, se describen los pasos a dar para generar conjuntos de pruebas que permitan la validación y evaluación de implementaciones de un protocolo o arquitectura de seguridad.

Sin embargo, la metodología no generará únicamente conjuntos de pruebas a llevar a cabo, sino que aspectos tales como los mecanismos para realizar las medidas, la interpretación de los resultados obtenidos en cada una de las pruebas y la inclusión de características que sin ser obligatorias son necesarias para utilizar el protocolo o arquitectura de seguridad son también definidos y especificados a partir de la metodología.

Todos estos factores hacen que la metodología propuesta genere conjuntos exhaustivo de pruebas que permitan obtener información pormenorizada acerca de la conformidad de una implementación con el estándar correspondiente, así como conocer las capacidades de rendimiento que dicha implementación puede ofrecer.

9.1.5. Aplicación de la metodología a la arquitectura IPsec

La aplicación de la metodología de validación y evaluación remota de implementaciones de protocolos de seguridad a la arquitectura IPsec ha generado un conjunto de pruebas que, diseñadas a partir de las guías y actuaciones definidas en la metodología, llevan a cabo una completa validación de la conformidad con el estándar y evaluación del rendimiento. En este desarrollo del conjunto de pruebas las aportaciones pueden clasificarse en tres tipos diferentes:

- Los conjuntos de pruebas en sí mismos
- Los métodos de de evaluación y recogida de información
- Los perfiles de tráfico definidos

En cuanto a los conjuntos de pruebas a llevar a cabo para cada uno de los factores de conformidad y rendimiento que es necesario evaluar, el conjunto de pruebas se estructura de la siguiente forma:

- Conformidad con el estándar
  - Conformidad criptográfica
  - Conformidad de los protocolos
  - Conformidad de los mecanismos de autenticación
  - Conformidad de los mecanismos de gestión de claves
  - Conformidad de otras herramientas y mecanismos
- Rendimiento
  - Ancho de banda
  - Máximo número de asociaciones de seguridad simultáneas
  - Capacidad de establecimiento de asociaciones de seguridad
  - Tiempo de proceso

Cada una de estas categorías consta de pruebas que han surgido del análisis de las características de la arquitectura de seguridad IPsec, estando diseñadas para obtener la información necesaria de forma eficiente a la par que eficaz. Parte de esta eficacia viene dada por los métodos de evaluación y recogida de la información propuestos, que han permitido adaptar cada prueba a las particularidades de IPsec, al tiempo que se optimizan los recursos necesarios para obtener la información deseada.

Por último, algunos de los perfiles de tráfico que se han definido permiten obtener información acerca del rendimiento máximo que puede ofrecer la implementación, lo que nos permitirá comparar diferentes implementaciones entre ellas, mientras que otros de los perfiles nos proporcionan información concerniente al rendimiento de la implementación en condiciones de la red y de tipo de tráfico a proteger concretas.

9.1.6. Plataforma de validación y evaluación remota de implementaciones de IPsec

Los desarrollos que se han llevado a cabo durante las diferentes fases de la presente tesis se han dividido en dos grupos claramente diferenciados: las implementaciones atómicas y la plataforma de validación y evaluación remota.

Las implementaciones atómicas son desarrollos especializados que, concentrándose en una prueba concreta del conjunto generado a partir de la metodología, permiten la ejecución de dicha prueba en unas condiciones

concretas, obteniendo la información que se había definido como objetivo de la prueba.

Estas pruebas atómicas han servido como evaluación de los procesos de análisis llevados a cabo, ya que las pruebas atómicas implementan los métodos de medición y captura de datos descritos en dichos procesos. Asimismo, las pruebas atómicas han permitido comprobar la completitud de la metodología a la hora de generar conjuntos de pruebas y métodos de evaluación. En el caso de que alguno de los métodos propuestos no fuese válido para obtener información al aplicarse a una situación real, el desarrollo de la prueba atómica nos permitiría detectar esta situación, facilitando la corrección del error.

Por otro lado, las pruebas atómicas también han permitido conocer problemas adicionales que deben afrontar las implementaciones de la metodología, y que, por lo tanto, serán aplicables a la plataforma de validación y evaluación remota. Estos problemas y condiciones que es necesario superar vienen dados también por el análisis de las diferencias entre la implementación de las pruebas atómicas y la plataforma de evaluación y validación remota. Este estudio nos presenta nuevos requisitos y restricciones que será necesario tener en cuenta al implementar la plataforma, y que ya se han tenido en cuenta a la hora de diseñar la plataforma.

Sin embargo, todas estas funciones desarrolladas por las pruebas atómicas no deben dejar de lado uno de los aspectos más importantes: las pruebas atómicas son desarrollos perfectamente funcionales de la metodología propuesta, y como tales permiten obtener información válida acerca de la conformidad con el estándar y el rendimiento de una implementación IPsec.

En cuanto a la plataforma de validación y evaluación remota, se han llevado a cabo numerosos análisis acerca de los nuevos retos que se presentarán para desarrollar el conjunto de pruebas resultante de la aplicación de la metodología a la arquitectura IPsec, proponiéndose soluciones a dichos problemas que servirán para minimizar el impacto de dichos retos en la utilidad de la plataforma.

Por lo tanto, podemos ver cómo los diferentes desarrollos que han surgido en esta tesis orientados a la plataforma de validación y evaluación remota, no sólo han servido para validar los análisis llevados a cabo anteriormente, sino que han permitido llevar a cabo un primer análisis de los requisitos y particularidades de las implementaciones que ha facilitado el diseño final y el desarrollo de la plataforma. Adicionalmente, estos desarrollos han proporcionado un conjunto de pruebas que abarcan todo el conjunto de pruebas propuesto, siendo capaces de ofrecer información acerca de todos los aspectos que se definen en la misma.

Adicionalmente, la utilización de la plataforma en varios proyectos de investigación ha servido como evaluación de la utilidad de la misma, así como

del interés existente en la industria actualmente por este tipo de herramientas. Por otro lado, el interés de miembros de la comunidad científica internacional por dicha plataforma y por la metodología en sí misma servirá para evitar el estancamiento del trabajo realizado.

9.2. Futuras Líneas de Investigación

Para finalizar, en esta sección se analizarán futuras líneas de investigación que surgen a partir de esta tesis doctoral, realizando un pequeño análisis acerca de cada una de estas líneas.

9.2.1. Aplicación a otros protocolos y arquitecturas de seguridad

Una primera línea de investigación que surge es la aplicación de la metodología a otros protocolos y arquitecturas de seguridad. Esta aplicación nos permitiría obtener conjuntos de pruebas que lleven a cabo los diferentes análisis de conformidad y rendimiento a implementaciones de múltiples protocolos y arquitecturas, lo que tendría una doble aportación. Por un lado, se podría contar con un conjunto de pruebas adaptado a cada protocolo y arquitectura, que, unido a los métodos de medición de información y captura de resultados nos permitirá contar con una recopilación de pruebas y métodos que servirán de base a muchos desarrollos futuros.

Por otro lado, al disponer de dicha recopilación de pruebas y métodos será posible llevar a cabo un estudio acerca de las particularidades de cada protocolo y arquitectura de seguridad, partiendo de las características reflejadas en los conjuntos de pruebas. Este análisis servirá de retroalimentación para los conjuntos de pruebas, que pueden beneficiarse de los resultados que se obtengan.

Una segunda fase de esta aplicación a otros protocolos y arquitecturas de seguridad nos permitirá comparar tanto los protocolos en sí mismos (cuáles ofrecen mayores problemas de compatibilidad, cuáles ofrecen mayor rendimiento), como cada uno de los componentes de seguridad de los que constan (mecanismos de autenticación, sistemas de gestión de claves, etc. . . .).

Por último la validación y evaluación de múltiples implementaciones de diferentes protocolos de seguridad utilizando los conjuntos de pruebas que se generen nos permitirán llevar a cabo el análisis y comparación de la conformidad y el rendimiento en múltiples implementaciones, incluyendo aquellas en dispositivos móviles o reducidos, que puedan presentar mayores limitaciones computacionales. De esta forma será posible llevar a cabo una comparativa que permita escoger entre aquellos protocolos y arquitecturas

de seguridad que mejor se adapten a nuestras necesidades.

9.2.2. Integración de la plataforma con otras herramientas

Otra línea de investigación que surge a partir de las aportaciones de la presente tesis es la integración de la plataforma de validación y evaluación remota con otras herramientas de gestión de la seguridad. Estas herramientas pueden ser otras metodologías de evaluación de la seguridad o incluso módulos de análisis de vulnerabilidades como los que se han analizado en el apartado 2.2.3 de esta tesis.

Otro aspecto de la integración de la plataforma con herramientas de gestión es el uso de mecanismos de configuración y gestión centralizada de los dispositivos de seguridad existentes en la red (como el propuesto para los dispositivos de redes privadas virtuales en [145]). Esta integración permitiría que las configuraciones recomendadas obtenidas de la plataforma sean directamente procesadas, traducidas al lenguaje de configuración de la implementación, e instaladas remotamente en los dispositivos en cuestión.

Al llevar a cabo algunas de estas integraciones es posible que surja la necesidad de aunar el diseño de las pruebas en la metodología, especialmente si dichas pruebas también surgen a partir de metodologías de análisis de la seguridad. En este caso, sería posible llegar a ampliar la metodología (y por extensión, los conjuntos de pruebas y la plataforma), incorporándose módulos a medida que sea necesario (como puede verse en la Figura 9.1). Esto es posible ya que la metodología contempla la posibilidad de llevar a cabo análisis específicos independientes, o integrados en conjuntos de pruebas que abarquen todas las áreas.

9.2.3. Interpretación de los informes

Una tercera línea de investigación que surge a partir de este proyecto es la de implementar mecanismos que permitan interpretar los informes generados por la plataforma, de forma que no se presenten únicamente los resultados de la realización de cada una de las pruebas. Algunos tipos de interpretación que pueden desarrollarse son los siguientes:

- Para las pruebas de conformidad, relación con el estándar, descripción del resultado esperado y del obtenido. Explicación de los resultados fallidos más comunes.
- Interpretación de los resultados de conformidad: ¿qué significan los resultados obtenidos?
- Relación con los resultados obtenidos por otras implementaciones del protocolo o arquitectura de seguridad: ¿con qué implementaciones y en qué condiciones puede interoperar la implementación evaluada?

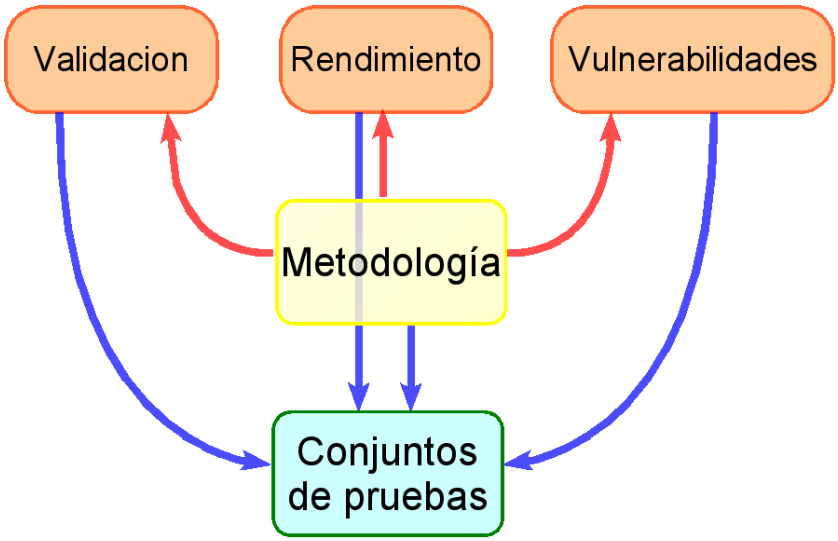

Figura 9.1: Posible diseño de la metodología al integrar el análisis de vulnerabilidades

- Comparación del rendimiento ofrecido por la implementación que es objeto del análisis con respecto a otras implementaciones analizadas anteriormente.

Esta interpretación tendría un punto clave en la elaboración de guías de configuración que optimicen la utilización de la implementación en función de los resultados de conformidad y rendimiento obtenidos. Estas guías resolverían preguntas como *¿Con qué configuración puedo establecer canales seguros con una mayor cantidad de implementaciones, al tiempo que maximizo mi rendimiento?*.

En una fase posterior, las guías de configuración podrían generarse dinámicamente, de forma que a medida que se introducen parámetros acerca de la infraestructura en la que dicha implementación debe operar, y otros requisitos de seguridad, rendimiento e interoperabilidad, se generen configuraciones de la implementación que permitan llevar a cabo la labor de protección de la información en el entorno que se describe.

Llegados a este punto, y si se ha conseguido la integración con herramientas de gestión y configuración que se ha mencionado en el apartado anterior, sería posible generar ficheros de configuración que se enviarían a las propias implementaciones, instalándose remotamente y preparando la implementación para operar bajo dicha configuración.

9.2.4. Estandarización de la metodología

Por último, un objetivo a largo plazo será la estandarización de la metodología y de los conjuntos de pruebas que se generen a partir de ella a través de algún organismo de estandarización. Como se comentó en los capítulos 1 y 2, en los últimos años el IETF ha procedido a estandarizar conjuntos de pruebas que permitan evaluar el rendimiento de diversos dispositivos de red. Por su parte, ISO e ITU-T cuentan con metodologías de análisis formales de la conformidad en sistemas abiertos, lo que indica que el interés de estos organismos por estas metodologías existe.

La estandarización, tanto de la metodología como de los conjuntos de pruebas que se generan al aplicarla a protocolos de seguridad concretos, permitiría responder a la llamada de la Unión Europea en el VII Programa Marco, proporcionando métodos, herramientas y mecanismos para facilitar la interoperabilidad entre sistemas de información e identificar aquellos problemas potenciales que puedan darse.

Adicionalmente, la aplicación de la metodología a protocolos y arquitecturas de seguridad podría incluirse en el proceso de diseño y estandarización de nuevos protocolos, de tal forma que junto con la especificación del protocolo se publiquen conjuntos de pruebas específicas que permitan a los desarrolladores y usuarios conocer si las implementaciones que tienen disponibles son conformes al estándar o no.

Bibliografía

[1] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz. Extensible authentication protocol (eap). Request For Comments RFC 3748, Internet Engineering Task Force, 2004.

[2] C. Adams. The cast-128 encryption algorithm. Request For Comments RFC 2144, Internet Engineering Task Force, 1997.

[3] C. Adams and J. Gilchrist. The cast-256 encryption algorithm. Request For Comments RFC 2612, Internet Engineering Task Force, 1999.

[4] C. Alberts and A. Dorofee. Octave^{SM} criteria. Technical Report CMU/SEI-2001-TR-016, ESC-TR-2001-016, National Institute of Standards and Technology, 2001.

[5] K. Anagnostakis, M. Greenwald, S. Ioannidis, A. Keromytis, and D. Li. A cooperative immunization system for an untrusting internet, 2003.

[6] R. J. Anderson. *Security Engineering: A Guide to Building Dependable Distributed Systems*. John Wiley & Sons, Inc., New York, NY, USA, 2001.

[7] G. Apostolopoulos, V. Peris, P. Pradhan, and D. Saha. Securing electronic commerce: reducing the SSL overhead. *IEEE Network*, 14(4):8–16, July 2000.

[8] G. Apostolopoulos, V. G. J. Peris, and D. Saha. Transport layer security: How much does it really cost? In *INFOCOM*, pages 717–725, 1999.

[9] R. Atkinson. Authentication header. Request For Comments RFC 1826, Internet Engineering Task Force, 1995.

[10] R. Atkinson. Ip encapsulating security payload (esp). Request For Comments RFC 1827, Internet Engineering Task Force, 1995.

[11] R. Atkinson. Security architecture for the internet protocol. Request For Comments RFC 1825, Internet Engineering Task Force, 1995.

[12] A. Balachandran, G. M. Voelker, P. Bahl, and P. Venkat Rangan. Characterizing user behavior and network performance in a public wireless lan. In *SIGMETRICS*, pages 195–205. ACM, 2002.

[13] G. Bartolomeo, F. Berger, H. J. Eikerling, F. Martire, and S. Salsano. Handling user profiles for the secure and convenient configuration and management of mobile terminals and services. In *DEXA Workshops*, pages 272–277. IEEE Computer Society, 2005.

[14] J. Beale. Bastille linux hardening program, 2006.

[15] H. Berkowitz, E. Davies, S. Hares, P. Krishnaswamy, and M. Lepp. Terminology for benchmarking bgp device convergence in the control plane. Request For Comments RFC 4098, Internet Engineering Task Force, 2005.

[16] V. Berzins. *Software Engineering with Abstractions.* Addison-Wesley Professional, Reading, MA, first edition, July 1991.

[17] O. Billet, H. Gilbert, and C. Ech-Chatbi. Cryptanalysis of a white box aes implementation. In H. Handschuh and M. Anwar Hasan, editors, *Selected Areas in Cryptography*, volume 3357 of *Lecture Notes in Computer Science*, pages 227–240. Springer, 2004.

[18] M. Bishop. *The Art and Science of Computer Security.* Addison-Wesley, Reading, Massachusetts, first edition, December 2001.

[19] S. Bradner and J. McQuaid. Benchmarking methodology for network interconnect devices. Request For Comments RFC 2544, Internet Engineering Task Force, 1999.

[20] Sonja Buchegger and Jean-Yves Le Boudec. Performance analysis of the confidant protocol. In *MobiHoc*, pages 226–236. ACM, 2002.

[21] OpenS/WAN bug tracking software. Opens/wan without nat-t crashes on nat-t conn, 2005.

[22] J. R. Burch, E. M. Clarke, and D. E. Long. Symbolic model checking with partitioned transistion relations. In *VLSI*, pages 49–58, 1991.

[23] C. W. Burrell. A methodology for planning and executing custom benchmarks. In *Int. CMG Conference*, pages 363–372. Computer Measurement Group, 1997.

[24] M. Burrows, M. Abadi, and R. M. Needham. A logic of authentication. *ACM Trans. Comput. Syst.*, 8(1):18–36, 1990.

[25] Ed C. Kaufman. Internet key exchange (ikev2) protocol. Request For Comments RFC 4306, Internet Engineering Task Force, 2005.

[26] M. Castelino and F. Hady. Tutorial on npf's ipsec forwarding benchmark. *Embedded.com*, October 2004.

[27] CERT Coordination Center. Unix configuration guidelines, 2006.

[28] CERT Coordination Center. Windows nt configuration guidelines, 2006.

[29] CERT. Multiple implementations of the radius protocol contain a digest calculation buffer overflow. Vulnerability Note VU#589523, US Computer Emergency Response Team, 2002. http://www.kb.cert.org/vuls/id/589523.

[30] CERT. Multiple implementations of the radius protocol do not adequately validate the vendor-length of the vendor-specific attributes. Vulnerability Note VU#936683, US Computer Emergency Response Team, 2002. http://www.kb.cert.org/vuls/id/936683.

[31] CERT. Multiple vendors' ssh transport layer protocol implementations contain vulnerabilities in key exchange and initialization. Vulnerability Note VU#389665, US Computer Emergency Response Team, 2002. http://www.kb.cert.org/vuls/id/389665.

[32] CERT. Multiple vulnerabilities in ssl/tls implementations. Vulnerability Note VU#104280, US Computer Emergency Response Team, 2003. http://www.kb.cert.org/vuls/id/104280.

[33] CERT. Check point isakmp vulnerable to buffer overflow via certificate request. Vulnerability Note VU#873334, US Computer Emergency Response Team, 2004. http://www.kb.cert.org/vuls/id/873334.

[34] CERT. Ieee 802.11 wireless network protocol dsss cca algorithm vulnerable to denial of service. Vulnerability Note VU#106678, US Computer Emergency Response Team, 2005. http://www.kb.cert.org/vuls/id/106678.

[35] S. Chow, P. A. Eisen, H. Johnson, and P. C. van Oorschot. White-box cryptography and an aes implementation. In K. Nyberg and H. M. Heys, editors, *Selected Areas in Cryptography*, volume 2595 of *Lecture Notes in Computer Science*, pages 250–270. Springer, 2002.

[36] Information Technology Committee. Information technology – framework – formal methods in conformance testing. ISO/IEC standard ISO/IEC 13245, International Organization for Standardization, 1995.

[37] Information Technology Committee. Information technology – open systems interconnection – conformance testing methodology and framework. ISO/IEC standard ISO/IEC 9646, International Organization for Standardization, 1995.

[38] Information Technology Committee. Information technology – open systems interconnection – lower layers security model. ISO/IEC standard ISO/IEC 13594, International Organization for Standardization, 1995.

[39] Information Technology Committee. Information technology – open systems interconnection – network layer security protocol. ISO/IEC standard ISO/IEC 11577, International Organization for Standardization, 1995.

[40] Information Technology Committee. Information technology – open systems interconnection – upper layers security model. ISO/IEC standard ISO/IEC 10745, International Organization for Standardization, 1995.

[41] Information Technology Committee. Information technology – security techniques – key management. ISO/IEC standard ISO/IEC 11770, International Organization for Standardization, 1995.

[42] Information Technology Committee. Information technology – open systems interconnection – security frameworks for open systems. ISO/IEC standard ISO/IEC 10181, International Organization for Standardization, 1997.

[43] LAN/MAN Standards Committee. Ieee standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements – part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications – amendment 2: Higher-speed physical layer (phy) extension in the 2.4 ghz band – corrigendum 1. IEEE Standard for Information Technology 802.11b, IEEE Computer Society, 2001.

[44] LAN/MAN Standards Committee. Ieee standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements – part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications – amendment 6: Medium access control (mac) security enhancements. IEEE Standard for Information Technology 802.11i, IEEE Computer Society, 2004.

[45] VPN Consortium. Vpnc testing for interoperability. Interoperatibility tests results, VPN Consortium, 2004.

[46] R. deGraaf, J. Aycock, and M. J. Jacobson Jr. Improved port knocking with strong authentication. In *ACSAC*, pages 451–462. IEEE Computer Society, 2005.

[47] L. Deri. Network top (ntop). Recurso Electrónico `http://www.ntop.org`, Centro Serra of the University of Pisa, 2005.

[48] T. Dierks and C. Allen. The transport layer security (tls) protocol version 1.1. Request For Comments RFC 2246, Internet Engineering Task Force, 1999.

[49] T. Dierks and E. Rescorla. The transport layer security (tls) protocol version 1.1. Request For Comments RFC 4346, Internet Engineering Task Force, 2006.

[50] Hans Dobbertin, Antoon Bosselaers, and Bart Preneel. Ripemd-160: A strengthened version of ripemd. In Dieter Gollmann, editor, *Fast Software Encryption*, volume 1039 of *Lecture Notes in Computer Science*, pages 71–82. Springer, 1996.

[51] J. J. Dujmovic, R. E. Kinicki, and S. Subbanna. Network benchmarking using distributed client/server pairs. In *Int. CMG Conference*, pages 183–194. Computer Measurement Group, 1995.

[52] J. J. Dujmovic, R. E. Kinicki, and S. Subbanna. A distributed client/-server benchmark for network performance measurement. *Informatica (Slovenia)*, 20(1), 1996.

[53] J. J. Dujmovic and H. Lew. A method for generating benchmark programs. In *Int. CMG Conference*, pages 379–388. Computer Measurement Group, 2000.

[54] D. Eastlake. Cryptographic algorithm implementation requirements for encapsulating security payload (esp) and authentication header (ah). Request For Comments RFC 4305, Internet Engineering Task Force, 2005.

[55] International Organization for Standardization. Information technology – open distributed processing – unified modeling language (uml) version 1.4.2. Standard 19501:2005, International Organization for Standardization, 2005.

[56] WAP Forum. Wireless application protocol architecture specification. Protocol Specification WAP-210-WAPArch-20010712, Wireless Application Protocol Forum, 2001.

[57] WAP Forum. Wireless transport layer security. Protocol Specification WAP-261-WTLS-20010406-a, Wireless Application Protocol Forum, 2001.

[58] A. Freier, P. Karlton, and P. Kocher. The ssl protocol version 3.0. Internet Draft 3.0, Netscape Communications, 1996.

[59] J. J. Garrett. Ajax: A new approach to web applications, 2005.

[60] R. Glenn and S. Kent. The null encryption algorithm and its use with ipsec. Request For Comments RFC 2410, Internet Engineering Task Force, 1998.

[61] T. J. Hacker, B. D. Athey, and B. Noble. The end-to-end performance effects of parallel tcp sockets on a lossy wide-area network. In *IPDPS*. IEEE Computer Society, 2002.

[62] A. Haley and S. H. Zweben. Development and application of a white box approach to integration testing. *Journal of Systems and Software*, 4(4):309–315, 1984.

[63] A. Harbitter and D. A. Menascé. Performance of public-key-enabled kerberos authentication in large networks. In *IEEE Symposium on Security and Privacy*, pages 170–183, 2001.

[64] A. Harbitter and D. A. Menascé. The performance of public key-enabled kerberos authentication in mobile computing applications. In *ACM Conference on Computer and Communications Security*, pages 78–85, 2001.

[65] D. Harkins and D. Carrel. The internet key exchange (ike). Request For Comments RFC 2409, Internet Engineering Task Force, 1998.

[66] R. Hefner. Lessons learned with the systems security engineering capability maturity model. In *ICSE*, pages 566–567, 1997.

[67] R. Hefner. A process standard for system security engineering: Development experiences and pilot results. In *ISESS '97: Proceedings of the 3rd International Software Engineering Standards Symposium (ISESS '97)*, page 217, Washington, DC, USA, 1997. IEEE Computer Society.

[68] P. Herzog. Open-source security testing methodology manual. Special Publication 2.1.1, Institute for Security and Open Methodologies, 2005.

[69] B. Hickman, D. Newman, S. Tadjudin, and T. Martin. Benchmarking methodology for firewall performance. Request For Comments RFC 3511, Internet Engineering Task Force, 2003.

[70] P. Hoffman. Cryptographic suites for ipsec. Request For Comments RFC 4308, Internet Engineering Task Force, 2005.

[71] R. Housley. Using advanced encryption standard (aes) ccm mode with ipsec encapsulating security payload (esp). Request For Comments RFC 4309, Internet Engineering Task Force, 2005.

[72] R. Housley, W. Polk, W. Ford, and D. Solo. Internet x.509 public key infrastructure certificate and certificate revocation list (crl) profile. Request For Comments RFC 3280, Internet Engineering Task Force, 2002.

[73] C. Hsu and U. Kremer. IPERF: A framework for automatic construction of performance prediction models. In *Workshop on Profile and Feedback-Directed Compilation (PFDC)*, 1998.

[74] M. Huth and M. Ryan. *Logic in Computer Science : Modelling and Reasoning about Systems*. Cambridge University Press, Cambridge UK, second edition, August 2004.

[75] International Systems Security Engineering Association (ISSEA). Systems security engineering capability maturity model sse-cmm model description document. Technical Report version 3.0, International Systems Security Engineering Association (ISSEA), 2003.

[76] ITU-T. Information technology - open systems interconnection - upper layers security model. ITU-T Recommendation X,803, Internet Telecommunication Union, 1994.

[77] ITU-T. Information technology - lower layers security model. ITU-T Recommendation X,802, Internet Telecommunication Union, 1995.

[78] ITU-T. Information technology - open systems interconnection - security frameworks for open systems: Overview. ITU-T Recommendation X,810, Internet Telecommunication Union, 1995.

[79] A. Izquierdo, J. Torres, J. M. Estévez, and J. C. Hernández. Attacks on port knocking authentication mechanism. In O. Gervasi, M. L. Gavrilova, V. Kumar, A. Laganà, H. Pueh Lee, Y. Mun, D. Taniar, and C. Jeng Kenneth Tan, editors, *ICCSA (4)*, volume 3483 of *Lecture Notes in Computer Science*, pages 1292–1300. Springer, 2005.

[80] G. P. Java. Iptraf - ip network monitoring software, 2005.

[81] R. Jones. Netperf: A network performance benchmark. White Paper Revision 2.1, Hewlett Packard, 1995.

[82] S. Kalidindi and E. Stewart. Ipsec virtual private networks: Conformance and performance testing. White paper, Ixia, 2003.

[83] P. Karn, P. Metzger, and W. Simpson. The esp des-cbc transform. Request For Comments RFC 1829, Internet Engineering Task Force, 1995.

[84] Vikas Kawadia and P. R. Kumar. Experimental investigations into tcp performance over wireless multihop networks. In *E-WIND '05: Proceeding of the 2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis*, pages 29–34, New York, NY, USA, 2005. ACM Press.

[85] S. Kent. Extended sequence number (esn) addendum to ipsec domain of interpretation (doi) for internet security association and key management protocol (isakmp). Request For Comments RFC 4304, Internet Engineering Task Force, 2005.

[86] S. Kent. Ip authentication header. Request For Comments RFC 4302, Internet Engineering Task Force, 2005.

[87] S. Kent. Ip encapsulating security payload (esp). Request For Comments RFC 4303, Internet Engineering Task Force, 2005.

[88] S. Kent and R. Atkinson. Ip authentication header. Request For Comments RFC 2402, Internet Engineering Task Force, 1998.

[89] S. Kent and R. Atkinson. Ip encapsulating security payload (esp). Request For Comments RFC 2406, Internet Engineering Task Force, 1998.

[90] S. Kent and R. Atkinson. Security architecture for the internet protocol. Request For Comments RFC 2401, Internet Engineering Task Force, 1998.

[91] S. Kent and K. Seo. Security architecture for the internet protocol. Request For Comments RFC 4301, Internet Engineering Task Force, 2005.

[92] T. Kivinen, B. Swander, A. Huttunen, and V. Volpe. Negotiation of nat-traversal in the ike. Request For Comments RFC 3947, Internet Engineering Task Force, 2005.

[93] D. Konz. A white box look at the performance of 802.11 wireless and its variants. In *Int. CMG Conference*, pages 285–302. Computer Measurement Group, 2004.

[94] M. Krzywinski. Port knocking: Network authentication across closed ports. *SysAdmin Magazine*, 12:12–17, 2003.

[95] R. Kuhn, T. J. Walsh, and S. Fries. Security considerations for voice over ip systems. Recommendation Guide 800-58, National Institute of Standards and Technology, 2005.

[96] B. K. Lee and L. K. John. Npbench: A benchmark suite for control plane and data plane applications for network processors. In *ICCD*, pages 226–233. IEEE Computer Society, 2003.

[97] C. A. Lee, J. Stepanek, R. Wolski, C. Kesselman, and I. Foster. A network performance tool for grid environments. In *Supercomputing '99: Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM)*, page 4, New York, NY, USA, 1999. ACM Press.

[98] H. E. Link and W. D. Neumann. Clarifying obfuscation: Improving the security of white-box des. In *ITCC (1)*, pages 679–684. IEEE Computer Society, 2005.

[99] Linux FreeS/WAN mailing list. Linux frees/wan compatibility guide. Interoperatibility tests results, Linux FreeS/WAN mailing list, 2004.

[100] V. Manral, R. White, and A. Shaikh. Ospf benchmarking terminology and concepts. Request For Comments RFC 4062, Internet Engineering Task Force, 2005.

[101] D. Maughan, M. Schertler, M. Schneider, and J. Turner. Internet security association and key management protocol (isakmp). Request For Comments RFC 2408, Internet Engineering Task Force, 1998.

[102] G. Memik, W. H. Mangione-Smith, and W. Hu. Netbench: A benchmarking suite for network processors. In *ICCAD*, pages 39–, 2001.

[103] P. Metzger and W. Simpson. Ip authentication using keyed md5. Request For Comments RFC 1828, Internet Engineering Task Force, 1995.

[104] S. Miltchev, S. Ioannidis, and A. D. Keromytis. A study of the relative costs of network security protocols. In C. G. Demetriou, editor, *USENIX Annual Technical Conference, FREENIX Track*, pages 41–48. USENIX, 2002.

[105] M. Minow. Minicomputer timesharing performance and usability. *SIGMINI Newsl.*, 4(3):4–16, 1978.

[106] NCSA. Ncsa mosaic history, 2000.

[107] C. Neuman, T. Yu, S. Hartman, and K. Raeburn. The kerberos network authentication service (v5). Request For Comments RFC 4120, Internet Engineering Task Force, 2005.

[108] P. Neumann. *Computer-Related Risks*. Addison-Wesley Professional, Reading, MA, first edition, October 1994.

[109] P. G. Neumann, R. J. Feiertag, K. N. Levitt, and L. Robinson. Software development and proofs of multi-level security. In *ICSE*, pages 421–428, 1976.

[110] NIST. Automated password generator. Federal Information Processing Standard 181, National Institute of Standards and Technology, 1993.

[111] NIST. Security requirements for cryptographic modules. Federal Information Processing Standard 140-1, National Institute of Standards and Technology, 1994.

[112] NIST. The transport layer security (tls) protocol version 1.1. Federal Information Processing Standard 186-2, National Institute of Standards and Technology, 2000.

[113] NIST. Advanced encryption standard. Federal Information Processing Standard 197, National Institute of Standards and Technology, 2001.

[114] NIST. Security requirements for cryptographic modules. Federal Information Processing Standard 140-2, National Institute of Standards and Technology, 2001.

[115] NIST. Automated security self-evaluation tool, 2004.

[116] NIST. Secure hash standard (shs). Federal Information Processing Standard 180-2, National Institute of Standards and Technology, 2004.

[117] NIST. National vulnerability database, 2006.

[118] NSA. Guías de configuración de la seguridad. Security Configuracion Guides-, National Security Agency, 2006. http://www.nsa.gov/snac/.

[119] Department of Defense. Trusted computer system evaluation criteria (orange book). Department of Defense Standard DOD 5200.28-STD, National Institute of Standards and Technology, 1985.

[120] National Institute of Standards and Technology. Ipsec web interoperability tester, 2000.

[121] National Institute of Standards and Technology. Common criteria for information technology security evaluation, part 1: Introduction and general model. CCMB-2005-08-001 version 2.3, National Institute of Standards and Technology, 2003.

[122] National Institute of Standards and Technology. Common criteria for information technology security evaluation, part 2: Security functional requirements. CCMB-2005-08-002 version 2.3, National Institute of Standards and Technology, 2003.

[123] National Institute of Standards and Technology. Common criteria for information technology security evaluation, part 3: Security assurance requirements. CCMB-2005-08-003 version 2.3, National Institute of Standards and Technology, 2003.

[124] National Institute of Standards and Technology. Nist net, 2005.

[125] Commission of the European Communities. Information technology security evaluation criteria. Council Recommendation version 1.2, Commission of the European Communities, 1991.

[126] Commission of the European Communities. i2010 a european information society for growth and employment. Technical report, Commission of the European Communities, 2005.

[127] Field Security Operations. Security review methodology. Security Technical Implementation Guide Version 2 Release 1, Defense Information Systems Agency, 2005.

[128] S. Owre, J. M. Rushby, and N. Shankar. Pvs: A prototype verification system. In D. Kapur, editor, *CADE*, volume 607 of *Lecture Notes in Computer Science*, pages 748–752. Springer, 1992.

[129] S. Pfleeger. *Software Engineering: The Production of Quality Software*. Macmillan Publishing Company, New York, USA, second edition, March 1991.

[130] J. Postel. Internet control message protocol. Request For Comments RFC 792, Internet Engineering Task Force, 1981.

[131] A. R. Prasad, P. Schoo, and H. Wang. An evolutionary approach towards ubiquitous communications: A security perspective. In *SAINT Workshops*, pages 689–695. IEEE Computer Society, 2004.

[132] K. Ramakrishnan, S. Floyd, and D. Black. The addition of explicit congestion notification (ecn) to ip. Request For Comments RFC 3168, Internet Engineering Task Force, 2001.

[133] M. Richardson and D. H. Redelmeier. Opportunistic encryption using the internet key exchange (ike). Request For Comments RFC 4322, Internet Engineering Task Force, 2005.

[134] R. Rivest. The md5 message-digest algorithm. Request For Comments RFC 1321, Internet Engineering Task Force, 1992.

[135] M. Roughan, S. Sen, O. Spatscheck, and N. G. Duffield. Class-of-service mapping for qos: a statistical signature-based approach to ip traffic classification. In A. Lombardo and J. F. Kurose, editors, *Internet Measurement Conference*, pages 135–148. ACM, 2004.

[136] J. Schiller. Cryptographic algorithms for use in the internet key exchange version 2 (ikev2). Request For Comments RFC 4307, Internet Engineering Task Force, 2005.

[137] B. Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. John Wiley & Sons, Inc., New York, NY, USA, 1995.

[138] Bruce Schneier. Description of a new variable-length key, 64-bit block cipher (blowfish). In Ross J. Anderson, editor, *Fast Software Encryption*, volume 809 of *Lecture Notes in Computer Science*, pages 191–204. Springer, 1993.

[139] Tenable Network Security. Nessus vulnerability scanner, 2006.

[140] A. Shacham, B. Monsour, R. Pereira, and M. Thomas. Ip payload compression protocol (ipcomp). Request For Comments RFC 3173, Internet Engineering Task Force, 2001.

[141] Q. Snell, A. Mikler, and J. Gustafson. Netpipe: A network protocol independent performance evaluator, 1996.

[142] JH. Song, R. Poovendran, J. Lee, and T. Iwata. The aes-cmac algorithm. Request For Comments RFC 4493, Internet Engineering Task Force, 2006.

[143] M. Swanson. Security self-assessment guide for information technology systems. Special Publication 800-26, National Institute of Standards and Technologies, 2001.

[144] Dyaptive Systems. Cdma network performance testing. White paper, Dyaptive Systems, 2003.

[145] J. Sánchez-Arévalo and J. M. Sierra. Sec-point : gestión segura de políticas de interconexión. Proyecto de fin de carrera, Universidad Carlos III de Madrid, 2004.

[146] Agilent Technologies. Validating ipsec network security devices. White Paper 598 9-1448EN, Agilent Technologies, 2004.

[147] B. Todd. Security auditor's research assistant, 2006.

[148] M. Tsai, C. Kulkarni, C. Sauer, N. Shah, and K. Keutzer. A benchmarking methodology for network processors, 2002.

[149] P. Wernick and M. M. Lehman. Software process white box modelling for feast/1. *Journal of Systems and Software*, 46(2-3):193–201, 1999.

[150] A. Whitten and J. D. Tygar. Usability of security: A case study. Computer Science Technical Report CMU-CS-98-155, Carnegie Mellon University, 1998.

[151] T. Wolf and M. Franklin. Commbench — a telecommunications benchmark for network processors, 2000.

[152] A. K. Y. Wong, T. S. Dillon, W. W. K. Lin, and M. T. W. Ip. M^2rt: A tool developed for predicting the mean message response time of communication channels in sizeable networks exemplified by the internet. *Computer Networks*, 36(5/6):557–577, 2001.

[153] D. K. Y. Yau, J. C. S. Lui, F. Liang, and Y. Yam. Defending against distributed denial-of-service attacks with max-min fair server-centric router throttles. *IEEE/ACM Trans. Netw.*, 13(1):29–42, 2005.

[154] T. Ylonen and C. Lonvick. The secure shell (ssh) transport layer protocol. Request For Comments RFC 4253, Internet Engineering Task Force, 2006.

[155] J. Yu, L. Jou, A. Matthews, and V Srinivasan. Criteria for evaluating vpn implementation mechanisms. Internet Draft-, Internet Engineering Task Force, 2000.

[156] J. K. Yun. Measuring network software performance. *Dr. Dobb's Journal*, 25(3):80, 82–91, Mar 2000.

[157] N. Ziring and S. Quinn. Specification for the extensible configuration checklist description format (xccdf) version 1.1.2. Interagency Report 7275, National Institute of Standards and Technologies, 2006.